September 2013

Vehicular Instrumentation and Data Processing for the Study of Driver Intent

Taha Kowsari
The University of Western Ontario

Supervisor
Dr. Steven Beauchemin
The University of Western Ontario

Graduate Program in Computer Science

A thesis submitted in partial fulfillment of the requirements for the degree in Doctor of Philosophy

Vehicular Instrumentation and Data Processing for the Study of Driver Intent

by

Taha Kowsari

Faculty of Science
Department Computer Science

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Abstract

The primary goal of this thesis is to provide processed experimental data needed to determine whether driver intentionality and driving-related actions can be predicted from quantitative and qualitative analysis of driver behaviour. Towards this end, an instrumented experimental vehicle capable of recording several synchronized streams of data from the surroundings of the vehicle, the driver gaze with head pose and the vehicle state in a naturalistic driving environment was designed and developed. Several driving data sequences in both urban and rural environments were recorded with the instrumented vehicle. These sequences were automatically annotated for relevant artifacts such as lanes, vehicles and safely driveable areas within road lanes. A framework and associated algorithms required for cross-calibrating the gaze tracking system with the world coordinate system mounted on the outdoor stereo system was also designed and implemented, allowing the mapping of the driver gaze with the surrounding environment. This instrumentation is currently being used for the study of driver intent, geared towards the development of driver maneuver prediction models.

Acknowledgements

I would like to show my greatest appreciation to my supervisor Steven Beauchemin. I cannot say thank you enough for his tremendous support and help. During the instumentation times he has been working like other members of the project in the hot and cold weather, and without his devotional help and guidance this porject would not have been successful.

I would like to thank other members of the team, espetially, Owen McCarthy, Ji Hyun Cho, Kevin Brightwell, and Mohsen Zabihi for their great work and help.

Last but not least I take this opportiunity to express my gratitude to my wife and my son, this project would have not been possible to complete without their support and patience.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Since August 1896 when Mary Ward killed in the first motor vehicle accident, 25 million people died in vehicle-related accidents. The proposed research program aims at reducing vehicle-related fatalities and injuries first by creating the required vehicular instrumentation for the study and modeling of driver intent.

Over recent years, auto manufacturers have introduced various systems to enhance driving safety, such as rear-view cameras and sensors, adaptive cruise controls, enhanced night vision, and so on. However, these systems (in production or under development in research laboratories) have yet to realize their full potential. Our envisioned integration of ADAS technologies reveals an extended capacity for context analysis and behavioural prediction that will enhance traffic safety [3].

The pervasiveness of vehicle-related injuries and casualties imposes enormous costs on our society. Such losses may become largely avoidable if addressed with proper technologies derived from research initiatives such as this one.

While there is large amounts of research performed on different aspects of ADAS systems such as lane departure and collision avoidance warning systems, little attention is devoted to the driver as a central part of the driving task. There is evidence that 95 percent of all accidents are caused by human error, and therefore monitoring and correcting driver behaviour seem to be crucial in the effectiveness of a future ADAS.

1.1 Literature Survey

A forecast from the World Health Organization states that among the 10 leading causes of disease burden[1], traffic-related injuries will rank 3rd in 2020, up from 9th in 1999 [21]. While injuries per driven kilometer are in decline in the developed countries [1], a reverse trend can be observed elsewhere in the world, especially in countries where car ownership is rising quickly. In addition, further significant gains in traffic safety in developed countries seem only possible via ADAS, since the impact of other forms of safety improvements have begun to plateau [7]. The functions one finds in various current ADAS are LDW (Lane Departure Warning), LCA (Lane Change Assistance), LKA (Lane Keeping Assistance), FCW (Forward Collision Warning), BSW (Blind Spot Warning), and DDW (Drowsy Driver Warning), among others. LDW, LCA, and LKA systems alert the driver when unintentional or unsafe lane changes are to take place, or to assist the driver in keeping or changing lanes. These systems determine the vehicle position with respect to detected lanes. This is usually performed with real-time computer vision algorithms, since the detection of road markings and curbs is essential to locate lanes. Additionally, such systems can be interfaced with vehicle turn signals, blind

[1]Disease burdens are measured in Disability Adjusted Life Years (DALYs).

spot warning systems, and possibly driver monitoring systems to determine if lane changes are safe and intentional. Stereo vision algorithms have been successfully adapted for some of these tasks [6, 28, 18, 17, 20] , supported by various computer vision techniques such as edge detection, image segmentation, feature tracking, and particle filtering. A consensus around vision-based techniques for lane detection has recently emerged [25] mainly because vision sensors provide accurate positional information which do not involve any instrumentation of highway infrastructures. When conditions prevent vision sensors to perform correctly, data from other sensing devices may be fused with the vision data to increase reliability. Other vision-based ADAS address specific aspects of driving. For instance, one finds methods and algorithms for detecting and evaluating the speed and distance of advance cars [42, 30], radar and vision-based adaptive cruise control systems [14, 26, 9, 36], driver and vehicle hardware in-the-loop frameworks [13, 29], driver attention and fatigue management [11], pedestrian detection [8, 12, 23, 10, 31], and collision warning systems [34, 38, 37, 4]. Advanced Driving Assistance Systems are generally designed to support decision making by providing ergonomic information on the driving environment, such as the presence of surrounding vehicles, potential hazards, and general traffic conditions. Sensing may be performed with radar [14, 41], lidar [15], or laser range finders [24]. However a majority of ADAS rely principally on vision systems supported by other sensor modes [25]. With such a variety of sensor modalities and hard real-time constraints, fusion becomes central to ADAS and the current literature reflects this fact in the large number of contributions that approach data and knowledge fusion in this context [19, 14, 9, 43]. Alternatively, several functions of ADAS may be realized using vehicle-to-vehicle (V2V) wireless communication protocols and Global Positioning Systems (GPS). Examples include the diffusion of traffic

information, [27, 35], collision warning systems [16, 39], lane changing assistance [2], and tracking neighboring vehicles [32]. While research on ADAS may integrate a number of different functions such as forward collision detection and lane departure tracking [22], most of them pay little attention to the monitoring of events and factors that are directly concerned with the driver of the vehicle. It is only recently that the cognitive aspects have been considered as a legitimate part of intelligent ADAS [33]. Since 95 percent of all accidents are caused by human errors such as cognitive (47 percent), judgment (40 percent), and operational errors (13 percent), it is crucial that these aspects of driving be a central part of any intelligent ADAS [40]. Keeping the driver as an active participant in the feedback mechanisms allows for providing contextually motivated informational support and offers immediate applications for enhancing safety [29]. Research efforts to integrate these various systems have very recently begun, on the premise that integration is likely to generate greatly enhanced functionality for the next generation of ADAS [3]. A survey of possibilities include the following types of integration:

1. Lane change assistance with vehicle detection, blind spot monitoring, driver head pose estimation, and car turn signal system: This type of integration allows for increased safety during lane change maneuvers as it is possible to determine the position and velocity of surrounding vehicles, while the driver head pose (looking into the blind spot) and monitoring of the turn signal systems allow to determine if the lane change is intentional.

2. Driver gaze direction monitoring with vehicle, obstacle, pedestrian, and road sign detection: This integration assists in determining if the driver is cognizant of the general traffic conditions surrounding the vehicle, and

brings the possibility of assessing whether the driver is seeing other vehicles, road signs, and obstacles in the visual field. Warnings may then be issued only if important traffic elements are not observed by the driver, preventing cognitive overload and unnecessary distractions, among other things. Additionally, the frequency with which the driver looks in the rear view mirrors, the instrumentation panel, and other odometry- related equipment can be determined and constitutes a significant factor when assessing the cognitive state of a driving agent.

3. Integrating vehicle odometry and sensor readings with computer vision systems of ADAS: Vehicle odometry obtained in real-time through the CANbus interface of the vehicle greatly improves and simplifies computer vision processes such as feature tracking and motion estimation. For instance, if the speed of the vehicle is known with accuracy, it is then possible to infer the motion that should be observed from the pavement surface and other elements of the scene. Any discrepancy with the inference would indicate that an obstacle is detected on the surface of the road, such as an advance vehicle. In this context it is relatively easy to deferentially determine the speed of this vehicle. Other benefits of integrating odometry with ADAS in general include the possibility to know the current state of the steering, brakes, and acceleration (positive or negative) of the vehicle, allowing once again to warn the driver only when the appropriate maneuvers are not being applied given the current driving context (driver level of attention, surrounding cars, and so on).

4. Integrating driver attention levels with the warning functions allows ADAS to make informed decisions regarding the mode and intensity with which warnings should be delivered to the driver. For example,

an audio warning may not be suitable when the driver is engaged in a conversation. Instead, seat or steering vibrations may be more effective in this case.

5. Integrating V2V communications with the current driving context information descriptors inside ADAS may substantially augment the knowledge level of traffic conditions prevailing beyond the range of visual sensors. As a result, warning on dangerous traffic conditions may be issued well before traffic problems enter the drivers field of view.

The resulting outcomes of such a level of integration and capability extensions of ADAS are many. For one, driving behaviour can be analyzed over time and correlated with traffic context and events to provide tailored warnings and to enable the prediction of driving actions in the short term. The benefits of observing a driver for extended periods of time include the assessment of driving patterns, delineating the most common errors, and performing driver re-training if so requested, as it was recently accomplished in the context of elderly drivers and driving simulators [5], and presently in instrumented vehicles. The extended possibilities of integrated, intelligent ADAS are very relevant research areas as they do not intend to replace the driver as much as to assist in the process of driving safely. As it has been pointed out by Petersson et al. [29], what remains to be automated to reach the state at which vehicles are completely autonomous in a practical manner turns out to be difficult and elusive in everyday driving situations. However driver support through i-ADAS research offers immediate applications that are socio-economically beneficial.

1.2 Research Overview

The primary goal of this research is to study driver intent towards the development of predictive models of driver actions. In this thesis, we describe a hardware and software vehicular instrumentation which permits the observation and recording of drivers in the context of urban driving with a minimum of interference due to the instrumented vehicle. Of particular interest, is to determine the relevance of ocular behaviour as it pertains to the next driving maneuver. The instrumented vehicle was designed in such a way as to provide sufficient contextual information to enable a comprehensive study of driver intent. In our context, driver intent is meant to signify the relationship between driver cephalo-ocular behaviour, current driving maneuvers, and vehicular attitude, with respect to the most probable next maneuver to be effected by the driver. Understanding this relationship is fundamental to the creation of predictive models of driver behaviour, that could be in turn, used in future ADAS. The central concept being that an ADAS with predictive capabilities could be more effective in protecting its driver if it could prevent a wrong maneuver before it is effected, as opposed to reacting to that maneuver, as it is applied by the driver.

The instrumented vehicle is capable of recording and computing the following information at a frame rate of 30 Hz:

1. video stream from a calibrated front stereo system. A depth map is automatically computed at frame rate.

2. video stream from a pair of rear and front panomorph lenses with a 360°field of view.

3. Vehicular information obtained from the internal network of the vehi-

cle[1]. This stream of data contains real-time information on important aspects of driving, including vehicle acceleration (in 3D), current speed, steering wheel angle, accelerator and brake pedal pressure, the state of turn signals, and more.

4. The vehicle also provides its own GPS coordinates, which are retrieved on its internal network.

5. A driver eye-tracker capable of providing 3D head pose and gaze detection data, together with other useful parameters such as saccade and blink frequencies.This system also provides confidence measures on the provided data.

6. An on-board software subsystem that relates the 3D driver gaze with the depth map from the front stereo sensors in a way as to provide the absolute 3D coordinates of the driver gaze into the environment.

The sum of these data channels and computations are recorded and computed at 30 Hz.

Many software components needed to be created in the process of instrumenting the vehicle:

1. The recorder: is an on-board computer system recording the sum of the data streams and performing the computation of the depth maps and the absolute 3D gaze coordinates in the reference frame of the front stereo system.

2. The calibration: allows the experimenters to calibrate the front stereo system and to cross-calibrate the eye-tracker with it.

[1]This is achieved through the CANbus interface.

3. The player: which allows experimenter to playback the recorded driving sequences for further analysis.

We used the instrumented vehicle to conduct experiments with 16 volunteer drivers under realistic driving condition[1]. The drivers followed a common path of 28.5 kilometers within the city of London, Ontario.

1.3 Contributions

This thesis is an inherent part of the RoadLAB research program, instigated by Professor Steven Beauchemin, and is entirely concerned with vehicular instrumentation for the purpose of the study of driver intent. Chapters 2, 3, 4, and 5 have been published (or in the process of being) in recognized peer-reviewed venues. In what follows I describe my contributions with regards to each publication within the thesis:

1. Chapter 2: S.S. Beauchemin, M.A. Bauer, T. Kowsari, and J. Cho, *Portable and Scalable Vision-Based Vehicular Instrumentation for the Analysis of Driver Intentionality, IEEE Transactions on Instrumentation and Measurement*, Vol. 61, No. 1, pp. 159-173, 2013.

    - Under a set of strict requirements, Professor Beauchemin gave me the responsibility of designing and implementing the hardware and software systems for the instrumented vehicle. This publication is a direct result of this effort.

2. Chapter 3: T. Kowsari, S.S. Beauchemin, and M.A. Bauer, *Map-Based*

[1]These experiments were conducted under Ethnic Approval Notice number 102849, between August 22 and September 28 2012

Lane and Driveable Area Detection, submitted to, VISAPP 2014 Conference, Lisbon, portugal.

- I designed and implemented a new method to detect lanes and driveable areas within them in an effort to provide the experimental data sequences with a reasonable annotation for further analysis. The detection of driveable areas within lanes in front of the experimental vehicle appears sufficient to determine in most circumstances if the current driver maneuver is correct.

3. Chapter 4: T. Kowsari, S.S. Beauchemin, and J. Cho, *Real-Time Vehicle Detection and Tracking Using Stereo Vision and Multi-View AdaBoost", IEEE Intelligent Transportation Systems Conference*, Washington DC, USA, pp. 1255-1260, October 5-7, 2011.

   - For the purpose of experimental data sequence annotation, I developed an approach to vehicle detection that uses the horizon plane to limit the number of generated hypotheses, sufficiently reducing the computational burden to achieve a 25Hz rate. This approach compared favorably with other recent methods found in the literature.

4. Chapter 5: T. Kowsari, S.S. Beauchemin, M.A. Bauer, D. Laurendeau, and N. Teasdale, *Multi-Depth Cross-Calibration of Remote Eye Gaze Trackers and Stereoscopic Scene Systems, accepted in IEEE Transactions on Instrumentation and Measurement, Dec. 2012.*

   - After initial discussions with Professor Beauchemin in which he delineated a general approach to implement the cross-calibration, I was able to design and successfully test a new method to relate the

3D driver gaze onto the stereo depth map of the front stereo system at frame rate (30Hz). This publication reports on the algorithms and experimental results.

1.4 Thesis Organization

The organization of the thesis is as follows, in Chapter 2 we introduce the instrumented vehicle, which is the main contribution of this thesis, and elaborate on the design and implementation of the hardware and software systems. In Chapters 3 and 4, contributions related to vehicle and lane detection are presented. In Chapter 3, our Ada-Boost-trained vehicle detection system used for detecting and annotating vehicles in the sequences is presented. In Chapter 4, a map-based lane detection system is devised and explained. Our lane detection system uses a particle filter-based tracker to fit spline lane sets extracted from the Google map of the drivers' path to the lane feature detected in the sequences. Using the stereo depth map, we also compute the safely drivable area in the lanes, which consists of the obstacle-free portion of the lanes ahead of the experimental vehicle.

In Chapter 5, another important contribution to this research is provided, which consists of a cross-calibration method for calibrating the gaze detection system with respect to the world coordinate system of the front stereo system. In this Chapter we provide an algorithm for back projecting the point of gaze on the image plane to find where the driver is looking at in absolute 3D coordinates of the front stereo system and at frame rate (30 Hz). Chapter 6 offers a conclusion and outlines paths for future research.

Bibliography

[1] National Highway Traffic Safety Administration. Traffic safety facts 2006: A compilation of motor vehicle crash data from the fatality analysis reporting system and the general estimates system. Technical Report HS 810 818, US DOT, 2006.

[2] S. Ammoun, F. Nashashibi, and C. Laurgeau. An analysis of the lane changing manoeuvre on roads : the contribution of inter-vehicle cooperation via communication. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 1095 –1100, june 2007.

[3] S. Ashley. Crashless cars: Making driving safer. *Sci. Am.*, December 2008.

[4] S. Atev, H. Arumugam, O. Masoud, R. Janardan, and N.P. Papanikolopoulos. A vision-based approach to collision prediction at traffic intersections. *Intelligent Transportation Systems, IEEE Transactions on*, 6(4):416–423, 2005.

[5] S.S Beauchemin, P.D. Varcheie, L. Gagnon, D. Laurendeau, M. Lavalliere, T. Moszkowicz, F. Prel, and N. Teasdale. Cobvis-d: A computer vision system for describing the cephalo-ocular behavior of drivers in a simulated driving context. In *International Conference on Image Analysis and Recognition (ICIAR)*, pages 604–615, Halifax, Canada, July 2009.

[6] M. Bertozzi and A. Broggi. Gold: A parallel real-time stereo vision system for generic obstacle and lane detection. *Image Processing, IEEE Transactions on*, 7(1):62–81, 1998.

[7] PT Blythe and AM Curtis. Advanced driver assistance systems: gimmick or reality? In *11th World Congress on ITS, Nagoya*, 2004.

[8] A. Broggi, M. Bertozzi, A. Fascioli, and M. Sechi. Shape-based pedestrian detection. In *Intelligent Vehicles Symposium, 2000. IV 2000. Proceedings of the IEEE*, pages 215–220. IEEE, 2000.

[9] D. Caveney, B. Feldman, and J.K. Hedrick. Comprehensive framework for multisensor multitarget tracking in the adaptive cruise control environment. In *International Symposium on Ad2 vanced Vehicle Control, Hiroshima, Japan*, 2002.

[10] M. Enzweiler, P. Kanter, and DM Gavrila. Monocular pedestrian recognition using motion parallax. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 792–797. IEEE, 2008.

[11] L. Fletcher, L. Petersson, and A. Zelinsky. Road scene monotony detection in a fatigue management driver assistance system. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 484–489. IEEE, 2005.

[12] D.M. Gavrila and S. Munder. Multi-cue pedestrian detection and tracking from a moving vehicle. *International journal of computer vision*, 73(1):41–59, 2007.

[13] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen. Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations. *Vehicle System Dynamics*, 44(7):569–590, 2006.

[14] U. Hofmann, A. Rieder, and ED Dickmanns. Radar and vision data fusion for hybrid adaptive cruise control on highways. *Machine Vision and Applications*, 14(1):42–49, 2003.

[15] A.S. Huang, D. Moore, M. Antone, E. Olson, and S. Teller. Finding multiple lanes in urban road networks with vision and lidar. *Autonomous Robots*, 26(2):103–122, 2009.

[16] J. Huang and H.S. Tan. Design and implementation of a cooperative collision warning system. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pages 1017–1022. IEEE, 2006.

[17] K. Huh, J. Park, D. Hong, D.D. Cho, and J.H. Park. Development of a vision-based lane detection system considering configuration aspects. *Optics and lasers in engineering*, 43(11):1193–1213, 2005.

[18] J. Kaszubiak, M. Tornow, RW Kuhn, B. Michaelis, and C. Knöppel. Real-time vehicle and lane detection with embedded hardware. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 619–624. IEEE, 2005.

[19] N. Kawasaki and U. Kiencke. Standard platform for sensor fusion on advanced driver assistance system using bayesian network. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 250–255. IEEE, 2004.

[20] T. Kowsari, SS Beauchemin, and J. Cho. Real-time vehicle detection and tracking using stereo vision and multi-view adaboost. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, pages 1255–1260. IEEE, 2011.

[21] E. Krug. Road traffic injuries. *WHO Overview Fact Sheet*, 2000.

[22] J.F. Liu, Y.F. Su, M.K. Ko, and P.N. Yu. Development of a vision-based driver assistance system with lane departure warning and forward

collision warning functions. In *Computing: Techniques and Applications, 2008. DICTA'08. Digital Image*, pages 480–485. IEEE, 2008.

[23] X. Liu and K. Fujimura. Pedestrian detection using stereo night vision. *Vehicular Technology, IEEE Transactions on*, 53(6):1657–1665, 2004.

[24] B. Ma, S. Lakshmanan, and A.O. Hero III. Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion. *Intelligent Transportation Systems, IEEE Transactions on*, 1(3):135–147, 2000.

[25] J.C. McCall and M.M. Trivedi. Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation. *IEEE Trans. Intell. Transp. Sys.*, 7(1):20–37, 2006.

[26] R. Möbus, M. Baotic, and M. Morari. Multi-object adaptive cruise control. *Hybrid Systems: Computation and Control*, pages 359–374, 2003.

[27] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. Trafficview: traffic data dissemination using car-to-car communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(3):6–19, 2004.

[28] S. Nedevschi, R. Schmidt, T. Graf, R. Danescu, D. Frentiu, T. Marita, F. Oniga, and C. Pocol. 3d lane detection system based on stereovision. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*, pages 161–166. IEEE, 2004.

[29] L. Petersson, L. Fletcher, and A. Zelinsky. A framework for driver-in-the-loop driver assistance systems. In *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, pages 771–776. IEEE, 2005.

[30] D. Ponsa, A. López, F. Lumbreras, J. Serrat, and T. Graf. 3d vehicle sensor based on monocular vision. In *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, pages 1096–1101. IEEE, 2005.

[31] M. Rapus, S. Munder, G. Baratoff, and J. Denzler. Pedestrian recognition using combined low-resolution depth and intensity images. In *Intelligent Vehicles Symposium, 2008 IEEE*, pages 632–636. IEEE, 2008.

[32] S. Rezaei, R. Sengupta, H. Krishnan, X. Guan, and R. Bhatia. Tracking the position of neighboring vehicles using wireless communications. *Transportation Research Part C: Emerging Technologies*, 18(3):335–350, 2010.

[33] M. Rockl, P. Robertson, K. Frank, and T. Strang. An architecture for situation-aware driver assistance systems. In *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pages 2555–2559. IEEE, 2007.

[34] N. Saunier, T. Sayed, and C. Lim. Probabilistic collision prediction for vision-based automated road safety analysis. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pages 872–878. IEEE, 2007.

[35] N. Shibata, T. Terauchi, T. Kitani, K. Yasumoto, M. Ito, and T. Higashino. A method for sharing traffic jam information using inter-vehicle communication. In *Mobile and Ubiquitous Systems-Workshops, 2006. 3rd Annual International Conference on*, pages 1–7. IEEE, 2006.

[36] M.A. Sotelo, D. Fernandez, J.E. Naranjo, C. Gonzalez, R. Garcia, T. de Pedro, and J. Reviejo. Vision-based adaptive cruise control for intelligent road vehicles. In *Intelligent Robots and Systems, 2004.(IROS*

2004). Proceedings. 2004 IEEE/RSJ International Conference on, volume 1, pages 64–69. IEEE, 2004.

[37] N. Srinivasa. Vision-based vehicle detection and tracking method for forward collision warning in automobiles. In *Intelligent Vehicle Symposium, 2002. IEEE*, volume 2, pages 626–631. IEEE, 2002.

[38] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(5):694–711, 2006.

[39] H.S. Tan and J. Huang. Dgps-based vehicle-to-vehicle cooperative collision warning: Engineering feasibility viewpoints. *Intelligent Transportation Systems, IEEE Transactions on*, 7(4):415–428, 2006.

[40] Y. Umemura. Driver behavior and active safety (overview), special issue driver behavior and active safety. *R&D Review of Toyota CRDL*, 39(2).

[41] G.R. Widmann, M.K. Daniels, L. Hamilton, L. Humm, B. Riley, J.K. Schiffmann, D.E. Schnelker, and W.H. Wishon. Comparison of lidar-based and radar-based adaptive cruise control systems. *Society*, 109(Part 7):126–139, 2000.

[42] J. Yi, S. Yang, I. Jang, J. Chung, and M. Lim. Automobile advance alarm system based on monocular vision processing. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 428–432. IEEE, 2007.

[43] Y. Zhu, D. Comaniciu, V. Ramesh, M. Pellkofer, and T. Koehler. An integrated framework of vision-based vehicle detection with knowledge fusion. In *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*, pages 199–204. IEEE, 2005.

Chapter 2

Instrumented Vehicle

This Chapter is a reformatted version of the following article:

S.S. Beauchemin, M.A. Bauer, T. Kowsari, and J. Cho, *Portable and Scalable Vision-Based Vehicular Instrumentation for the Analysis of Driver Intentionality, IEEE Transactions on Instrumentation and Measurement*, Vol. 61, No. 1, pp. 159-173, 2013.

A portable and scalable vehicular instrumentation designed for on-road experimentation and hypothesis verification in the context of designing i-ADAS prototypes is described in this contribution.

2.1 Introduction

World-wide deaths from injuries are projected to rise from 5.1 million in 1990 to 8.4 million in 2020, with traffic-related injuries representing the major cause for this increase [15, 1]. Our research aims at reducing these fatalities by first developing a deeper understanding of the cognitive (cephalo-ocular) task of driving, identifying related risk factors and integrating these findings into predictive models of driver intentionality. The long term goals of this program in-

clude the identification of the cognitive factors involved in driving that impact traffic safety, the definition of sound principles for the design of automated vehicular safety technologies, and the development of intelligent, Advanced Driving Assistance Systems (i-ADAS), with driver behavior prediction and correction as the central tenet of safety improvement.

While research on ADAS may integrate a number of different functions such as forward collision detection and lane departure tracking [21], little attention is devoted to the monitoring of events and factors that directly concern the driver of the vehicle. It is only recently that cognitive aspects have been considered as a legitimate part of intelligent ADAS [25]. Since 95 percent of all accidents are caused by human error, it is crucial that these aspects of driving be a central part of intelligent ADAS [31]. Keeping the driver as an active participant in the feedback mechanisms allows for providing contextually motivated informational support and offers immediate applications for enhancing safety [23].

The extended possibilities of integrated, intelligent ADAS are very relevant research areas as they do not intend to replace the driver as much as to assist in the process of driving safely. As it has been pointed out by Petersson *et al.* [23], what remains to be automated to reach the state by which vehicles become completely autonomous in a practical manner turns out to be difficult and elusive in everyday driving situations. In light of this, it is our belief that driver support through i-ADAS can be deployed more readily, with consequent socio-economic benefits.

This contribution rests on earlier work in which preliminary instrumentation and tests were recently conducted [6]. It however differs significantly in that it motivates the instrumentation in the form of clearly stated hypotheses derived from a central conjecture and provides a performance evaluation

of the platform, along with identified physical, sensory, and computational limitations.

Our approach, while sharing common elements with those of others, is unique in several ways. First, we designed a portable instrumentation requiring no modification to the vehicular platform, using low-cost off-the-shelf components that are widely available. Second, our on-board computational approach rests on scalability. That is to say, additional computing power can easily be added to the current instrumentation, without any modifications to the existing system. This of course is a core requirement, as algorithms must be run in real-time. Third, our approach integrates the driver in the system as an inherent behavioral agent, in the aim of understanding and predicting driving actions.

2.2 Related Literature

While many research groups provide brief descriptions of their vehicular instrumentation in the context of driving assistance, such as [22, 10] for vision systems, and [28, 11] for multi-sensor instrumentation, few contributions directly address instrumentation strategies, concepts, and implementation in the context of ADAS. A notable exception is by Thrun [30] in the context of autonomous driving in which the sensory interface, perception modalities, planning and control, user interfaces, and software services are described in extensive detail. The motivation for our contribution partly stems from the observation that the related literature is currently sparse.

2.3 Hypothesis Verification

Our primary goal is to determine whether driver intentionality and driving-related actions can be predicted from quantitative and qualitative analyses of driver behavior. This broad question, while only partially answered [18], conveys its importance in more than one way. For instance, predictive formulations of the cognitive aspects of driving open the way to the design of reliable models of driver intentionality prediction and may lead to advances in safety-related vehicular technologies and accident prevention strategies from the perspective of on-board safety systems, up to assisting in guiding policies regarding the regulation of i-ADAS.

2.3.1 Primary Conjecture

Studies of driver behavior have approached the problem of intentionality from various perspectives. Driver behavior models have been suggested along with their empirical validations, with varying degrees of success [19]. However, *"the most effective technology may be that which monitors driver state and driving behavior in order to help attend to the roadway and recognize unsafe behavior"* [9]. In addition, it has been demonstrated time and time again that eyes, in general, *"look directly at the objects they engage with"* [3] and that the *"fixation that provides the information for a particular action immediately precedes that action"* [4]. These observations support our primary conjecture, which states that *if one considers a vehicle as an extension to the inherent human capability for motion, then one must also admit the possibility that eye movements are as predictive of driving actions as they are of physical movement.* The underlying rationale from which our conjecture stems rests on the demonstration that eye movements reflect processes aimed at locating the information required to

generate actions in relation to the environment [13, 16, 18].

2.3.2 Hypotheses

Since 95% of all accidents are caused by human error, it is imperative that drivers be the central element of systems that provide driving support [32]. Consequently, our short term goals consist of the empirical testing of hypotheses derived from the primary conjecture, in the hope of demonstrating that on-board vehicle safety systems which focus on the predictability of driver behavior are capable of significantly increasing driving safety. Toward this end, our primary conjecture is functionally fragmented into a number of hypotheses which can be investigated effectively and objectively:

1. *Cephalo-ocular behavior correlates with driver intentionality and precedes driving actions:* This hypothesis has been demonstrated in certain driving circumstances, as it is known that drivers negotiating a bend fixate on its tangent point to gather information on its curvature. This fixation precedes steering adjustments by an average of 0.8s [16]. Are there other driving circumstances (negotiating intersections, merging, highway driving, etc.) for which particular ocular behavior precede driving actions? While it is clear that ocular behavior cannot be constantly predictive of actions due to secondary drivers tasks (such as attending to vehicle functions), it is important to determine which behaviors possess a predictive value. However, the possibility exists that ocular behavior may not be sufficiently correlated with intentionality for use in prediction models. In this case, other investigative avenues may be possible, particularly through the observation of maneuvers being applied to the vehicle by the driver. Current actions may be predictive of future actions

and determining to which extent this may be the case would be central to this scenario. A third investigative avenue may be that both current driver maneuvers and ocular behavior are sufficient for useful prediction purposes.

2. *Driver levels of attention are indicative of the meaningfulness of cephalo-ocular behavior:* Driver visual attention is a central part of safe driving. It has been shown that driver glances away from the road for 2s or more resulted in 3.6 times more involuntary lane departures than glances of 1s [20]. Conversely, long visual fixations are not necessarily synonymous with attention. While eyes may be fixating, attention may not be elicited by events in the visual field. However, certain ocular patterns such as fixations accompanied by regular saccades are descriptive of the visual search behavior for information acquisition processes and correlate with drivers attending to the roadway [26]. The identification of factors providing indications of meaningful cephalo-ocular movements is necessary in order to assess whether or not the ocular behavior represents intent.

3. *Information delivered to drivers does not increase their cognitive loads:* Drivers are exposed to increasing flows of information provided by modern on-board vehicle functions. Recent studies have revealed that drivers are not always capable of eliciting a correct response to such solicitations due to, among other factors, the complexity of the driving context, or an increased cognitive load generated by actions not directly related to driving [2]. While it is suspected that the aforementioned hypothesis does not hold in general, it is crucial to experimentally determine the modalities, timings, and types of delivered information that can be tolerated and understood sufficiently rapidly by drivers, such that there is available

time to perform corrective maneuvers [9]. Still, it can be conjectured that in most circumstances the cognitive loads of drivers may already be high when safety-related information must be issued, probably increasing driving risk rather than reducing it. In the case this conjecture proves correct, it may become fruitful to investigate automated driving interventions (without delivery of information) in particularly demanding traffic contexts, or when information would not come in time, or otherwise distract drivers even more.

4. *Visual stimuli drivers attend to can be identified:* Salient elements in the visual field of drivers elicit cephalo-ocular responses aimed at attending to such stimuli. Correspondences between cephalo-ocular behavior and visual stimuli must be established to identify the elements within the visual field to which driver attention is turned. This knowledge will allow predictive models to assess whether or not drivers are attending to the appropriate stimuli, given current traffic contexts. This requirement implies that elements in the environment be correctly identified, located, and intersected with the 3D gaze direction of the driver. Consequently, systems in charge of processing the output of stereo sensors must reliably detect the presence of other vehicles, pedestrians, and obstacles in general. This objective has only been partially attained with the use of passive sensing (CCD cameras) mainly because the reliability of most (if not all) techniques greatly depends on visual scene conditions [33]. While it is expected that passive vision systems will fail from time to time in difficult driving conditions, there may be effective methods of providing enhanced reliability by way of combining other sources of vehicular information. Vehicle-to-Vehicle (V2V) inter-communication may

be used in situations where vision systems fail or under-perform, such as times when fog, snow, or rain are present. Such communication modalities have the potential to both enrich and extend the range of visual sensors when surrounding vehicles signal their presence and position. These ideas may enhance the robustness of on-board vision systems and are further investigated in [5].

The creation of effective predictive driving behavior models rests on the confirmation of these hypotheses. While it is not expected that every aspect of these ideas can be empirically demonstrated, it is believed their investigation will extend the current knowledge of the cognitive task of driving and allow for the establishment of strong principles for the design and operation of future on-board safety systems.

2.4 Layered Approach to Vehicular Instrumentation

The next generation of i-ADAS will require extensive data fusion and analysis processes owing to an ever increasing amount of available vehicular information. In this context a layered approach is best suited for real-time processing. In particular, such an approach enables bringing real-time data from sensors to a common level of compatibility and abstraction which significantly facilitates fusion and analysis processes. Our proposed computational model consists of four layers, with increasing levels of data abstraction (see Figure 2.1). The innermost layer consists of the hardware and software required to capture vehicle odometry, sequences from visual sensors, and driver behavioral data. The second layer pertains to hardware synchronization, calibration, real-time data

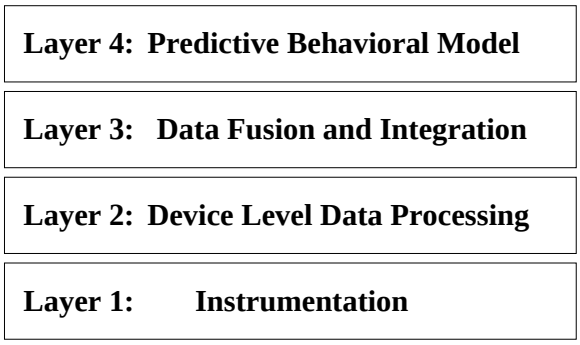

Figure 2.1: *The four layers comprising the data processing strategy on-board the instrumented vehicle.*

gathering, and vision detection processes. The third layer is where the data is transformed and fused into a single 4-dimensional space (x, y, z, t). The last layer makes use of the fused data to compare driver behavioural data with models of behavior that are appropriate given current odometry and traffic conditions. While we proceed to describe the four layers, it is to be noted that this contribution specifically addresses the instrumentation (layers one and two) and its performance evaluation.

2.4.1 Instrumentation

Contemporary vehicles equipped with On-Board Diagnostic systems (OBD-II) allow vehicle sensors to report on current status, and constitute the interface through which odometry is made available in real-time. Since 2008, the CAN-bus protocol[1] has become mandatory for OBD-II. This standardization simplifies the real-time capture of vehicle data. OBD-II to USB hardware interfaces with appropriate drivers are now common devices used to feed vehicle-related information to on-board computers or similar devices. The available informa-

[1]The CANbus (Controller Area Network bus) provides micro-controllers with the means to communicate with each other within a vehicle.

tion relevant to i-ADAS applications include current speed and acceleration (longitudinal and lateral), steering wheel rotation, state of accelerator and brake pedals, and independent wheel speed, which are real-time data captured at frequencies generally comprised between 20 and 200Hz. These elements provide the information that is required to understand the maneuvers effected by the driver.

In addition, several vision systems must instrument the vehicle in order to appropriately monitor the immediate environment (lanes, other vehicles, pedestrians, obstacles, etc) and the behavior of the driver (gaze direction, level of attention, etc). These hardware systems must be capable of high sampling rates (30Hz or more) such that sufficient accuracy in image processing and automated vision processes is achieved. It is useful to keep in mind that the position of a vehicle moving at 120 kph changes by 33 meters every second.

Similar observations apply concerning the changes in visual gaze direction (known as saccades) as they occur very rapidly. For this reason, vision hardware monitoring the gaze direction of the driver must have sufficiently dense sampling rates as to allow for deriving driver intentionality prior to the execution of the anticipated behavior [17]. This part of the vehicle instrumentation is realized with commercial hardware and software[1] from which data such as eye gaze direction, vergence distance, and saccade events are obtained at a frequency of 60Hz.

Also part of the instrumentation layer is a GPS device which is used by Vehicle-to-Vehicle (V2V) communications systems to provide other near-by instrumented vehicles with knowledge of traffic conditions beyond the range of their visual sensors.

Last but not least, on-board computing capabilities must also be sufficient

[1]FaceLAB 5^{TM} implements our instrumentation for eye tracking.

to process the sum of incoming data in real-time. To this end we have designed and assembled a computer for real-time data processing and fusion consisting of 16 cores, each running at 3.0GHz, with 16GB of internal memory and a 128GB Solid State Drive (SSD), with Linux Debian 5.01 as the operating system. The nodes are networked with a high-end gigabit network switch, and configured as a disk-less cluster, with the master node providing the operating system image to other nodes.

2.4.2 Justification of sensors selection

In the context of our hypothesis, it is vital that the instrumentation be able to provide information on current (and expected) driver behaviour and vehicle operation. Two subsystems contribute to this goal. First, an OBD-II to USB interface[1] sends vehicular data (odometry and vehicle operation) to the on-board computer for recording or real-time analysis, or both. Second, an eye and head pose tracking system provides the necessary data for cephalo-ocular behavior recording and analysis. The sum of these subsystems provide the information required to determine the interactions between the driver and the vehicle, in addition to ocular behavior parameters. The resulting instrumentation allows to identify the visual stimuli drivers respond to in relation with the driving surroundings and the most probable behavior to be observed next.

Our choice of passive sensors is motivated by the fact that data acquisition is non-invasive and provides information conveyed by visual elements such as road markings and signs, which are critical to the task of driving and yet unavailable to active sensors such as radars or range finders [7]. In addition, multiple lens and sensor configurations are possible. For instance, IR or near-

[1] A Kvaser Leaf Light^{TM} OBD-II to USB device implements this part of the instrumentation.

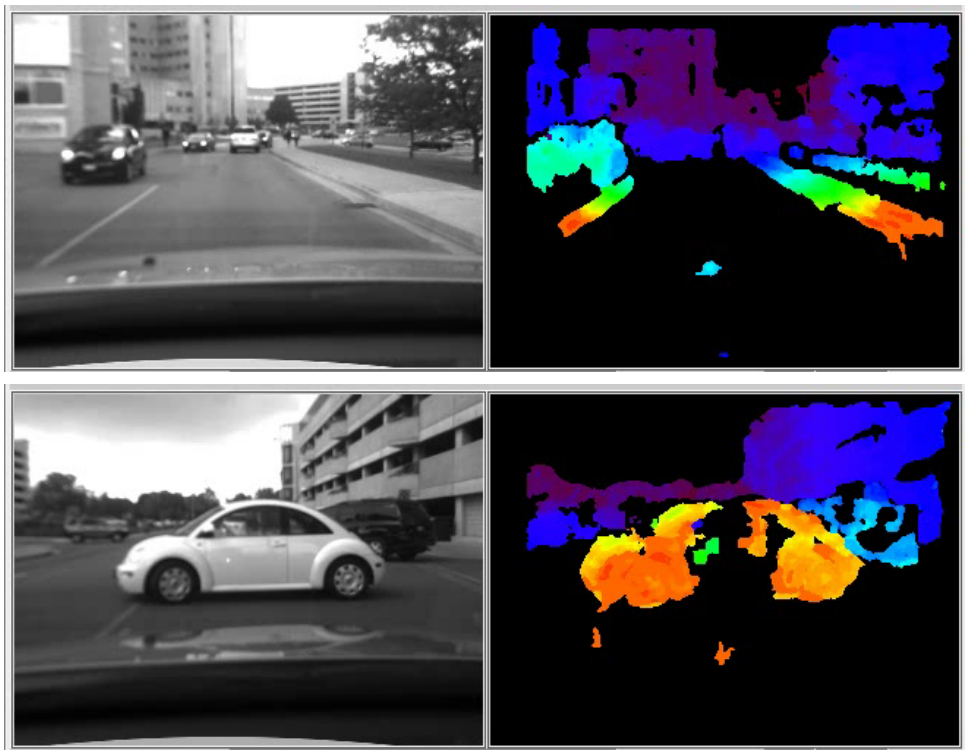

Figure 2.2: *Color-coded calibrated stereo depth maps are obtained at 30Hz. The distance between the instrumented vehicle and the roadside curbs, and other vehicles, is estimated in real-time.*

IR (Infra-Red) filters or sensors may readily be installed to provide night vision. Conversely, lenses of various types may be mounted on the sensors without any design modifications.

2.4.3 Device-Level Data Processing

For visual sensors, it is critical to obtain precise calibration parameters such as lens distortion, the optical center, and the external orientation of sensors with respect to each other. This calibration is required to perform stereo and to estimate distances of objects (other vehicles, pedestrians, etc.), which in turn greatly simplifies other vision-related tasks such as estimating motion, track-

ing, and obstacle detection. The RoadLAB stereo calibration interface was designed for this process. The interface is implemented using a calibration algorithm from the OpenCV 2.1 open source library based on Zhang's technique [34]. The calibration process consists of two steps. Intrinsic parameters are first estimated for each sensor and then, based on these, the extrinsic parameters for all possible sensor pairs are obtained. It is also possible to estimate the extrinsic parameters dynamically [8]. All the image frames from visual sensors are synchronized to within 125 μs. Once the synchronized frames are obtained, stereo depth maps are computed at frame rate, based on the calibration parameters (see Figure 2.2). The GPS data is obtained through `gpsd`, a GPS service daemon from `http://gpsd.berlios.de/` which provides an event-driven architecture. The data from the OBD-II/CANbus is obtained in a similar manner by creating a software layer for this purpose. Additionally, the incoming data from the instrumentation provides timestamps, allowing the system to fuse and select data elements in a synchronized fashion.

2.4.4 Data Fusion and Integration

Streams of data and video frames coming from monitoring the driver, the environment, and vehicle odometry must be placed in a suitable context for use by the behavioral prediction engine. We define a driver-centered frame of reference, in which elements of the Cognitive State of Driver (CSD) descriptor (head pose, gaze direction, blink events, lip movement), the Contextual Feature Set (CFS) descriptor (road lanes, other vehicles, pedestrians, etc), and the Vehicle State of Odometry (VSD) are transformed into, from their local sensor frames of reference (see Figure 2.3 for a depiction of the CSD and CFS descriptors in the context of our layered model). This is performed by using

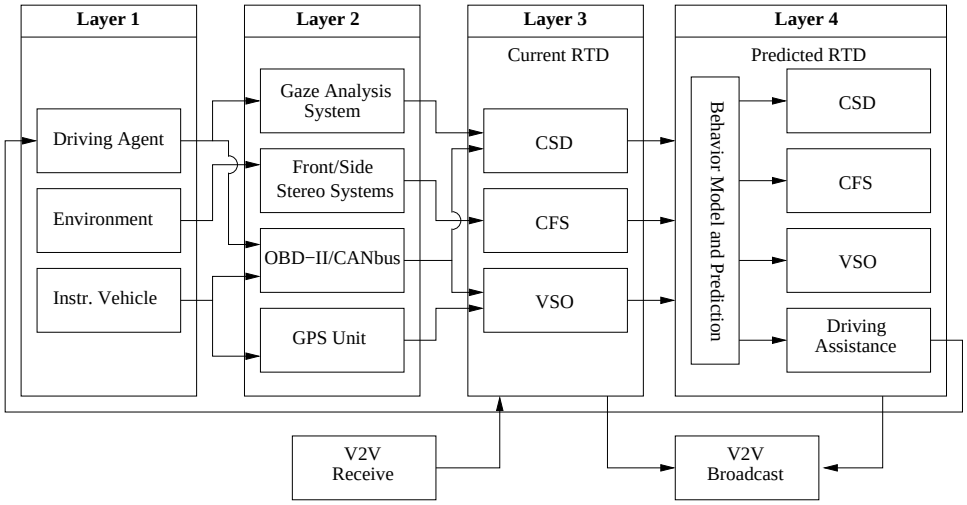

Figure 2.3: *A description of the retroactive mechanism operating between the current and predicted RTDs with respect to the outlined layered approach, in which driving assistance impacts both the current and predicted behavioral state of the driving agent. The reception of V2V information enriches the current CFS, which in turn impacts the predicted RTD. Informational elements from both the current and predicted RTDs are broadcast to other instrumented vehicles.*

the extrinsic parameters obtained with the calibration of the visual sensors with respect to each other. With these elements fused into a single frame of reference, the current CSD, CFS, and VSO descriptors are updated at 30Hz, and made available to the behavioral prediction engine.

Two modes of operation exist at this level. A recording mode captures the data and video streams from the instrumentation for in-laboratory, off-line analysis. A processing mode which performs as an i-ADAS operating in real-time is also possible. Each sequence generated for off-line analysis obeys a strict format standard, in which the calibration data, the timestamped frames from the stereo systems, and the vehicle odometry are recorded at 30Hz.

2.4.5 Predictive Behavioral Model

Our general hypothesis stems from research demonstrating that eye movements reflect moment-to-moment cognitive processes used to locate the information needed by the motor system for producing actions in relation to the environment [17, 14, 24]. This hypothesis is the foundation for our conjecture stating that the analysis of driver gaze direction (and other facial features) fused with the knowledge of the environment surrounding the vehicle (and its odometry) lead to the possibility of predicting driving behavior for short time frames (a few seconds). To accomplish these goals, it is necessary to infer a behavioral driving agent model that puts in relation the cognitive state of the driver, the vehicle odometry, and its surrounding environment as captured by sensors. For this purpose, we devise a Real-Time Descriptor (RTD) for a moving vehicle essentially consisting of a CFS, a CSD, and a VSO descriptor.

These elements represent the knowledge required in composing an extensive RTD suited for our purposes. While we are interested in deriving practical and predictive driving agent models, it is worth noting that both the CFS and the VSO possess predictive models which are less difficult to formulate. We further propose to structure the elements of the RTD within a retroactive mechanism (see Figure 2.3) in which both the current and predicted descriptors (CSD, CFS, and VSO) assist in determining not only the safety level of the context derived from the current RTD, but also that posed by the predicted RTD.

At the heart of the behavioral prediction engine is a Bayesian model which takes the current CSD, CFS, and VSO as inputs and predicts actuation behavior of the driver in the next few seconds. It also gathers statistical information about driving decisions and errors in a Driver Statistical Record (DSR) which can be used over time to improve the prediction accuracy. The current CSD

and CFS are in turn used to establish a Driver Memory of Surroundings (DMS) based on the attention level and gaze direction analysis of the driver. A General Forgetting Factor (GFF) is applied to the DMS as time elapses to reflect common characteristics of short-term visual memory. In addition, a Driver Cognitive Load factor (DCL) is inferred, based on the activities engaged by the driver, which in turn impacts the DMS, among other things.

2.5 Inter-Vehicular Communication

Vehicular networks have been an area of research for the past two decades [27]. Interest has been shown by researchers, government agencies, and automobile manufacturers in developing the technologies and protocols for vehicular networks. There is a number of major areas of interest where unique problems must be solved, including protocols for the physical and link layer, higher layer protocols to deliver traffic, safety, and security information. Beyond these purely technical challenges, vehicles created by different manufacturers must be able to communicate, thus rendering standardization essential.

Some standardization has already occurred with the IEEE 802.11p draft standard and allocation of 75MHz in the 5.9GHz spectrum for Dedicated Short Range Communications (DSRC) for the physical and link layer protocols. Further standardization with the IEEE 1609 draft standards for higher level protocols and services is ongoing [12]. Nonetheless, there are numerous open areas of research where solutions must be found before vehicular networks are adopted in consumer vehicles. How these two sets of technologies intersect is a topic that currently has not been looked at in depth. Vehicular networking technologies can provide detailed information about other vehicles in a large area, while sensor-based technologies can provide more detailed information

about the environment immediately surrounding a vehicle in real-time. How these two sets of technologies intersect is a topic that currently has not been looked at in depth.

Being able to combine both sources of information provides greater detail and breadth than any one technology can provide on its own. How to do this exactly is an open research problem. There is also no guarantee that the information provided by these separate system will agree. There is a wide variety of circumstances in which data from both systems may not match and the vehicle will need to deduce which one is most likely correct.

Our approach consists of the systems, strategies, and implementation of the concept of using V2V to extend on-board visual sensor range. Coupling V2V and sensory input may increase detection reliability and range for visual sensors. Conversely, sensors may inform i-ADAS of the presence of non-communicating elements such as pedestrians and non-vehicular obstacles. The potential that is held by integrating V2V communication with on-board sensory perception is considerable.

An instrumented vehicle navigating in an environment where other vehicles are similarly equipped would have access to critical traffic information well beyond the range of its sensors. Additionally, cascading information between communicating vehicles would allow a single vehicle to decide upon the range within which traffic information is deemed desirable.

While it seems natural to integrate sensory information with V2V in the context of i-ADAS, few research efforts have been conducted toward this goal. We believe that the complementarity of information obtainable from on-board sensors and V2V communication can form the basis for new approaches in driver-assisted systems and technologies [5].

Figure 2.4: *The RoadLAB in-vehicle laboratory:* **a) (left):** *on-board computer and LCD screen,* **b) (center):** *dual stereo front visual sensors,* **c) (right):** *side stereo visual sensors.*

2.6 In-Vehicle Laboratory

The design of the instrumented vehicle follows principles of sensor portability and computing scalability. Sensor portability is achieved by using vacuum devices to attach the instrumentation equipment to the the interior or exterior surfaces of the vehicle (see Figure 2.4), such as stereo camera rigs and LCD displays. Similarly, computing scalability is addressed with a disk-less, master-slave cluster of computing nodes, configured with open source software from Sandia National Laboratories (OneSIS). Additional computing nodes and Graphical Processing Units (GPUs) may be added at will, with the obvious cargo limitation imposed by the instrumented vehicle. Portability enables the use of a wide variety of vehicles without compromising their physical integrity, while computing scalability ensures an adequate supply of processing cores, matching the many possible sensor configurations (see Figure 2.5).

2.6.1 Physical Equipment

Each minute, the sensory equipment sends 2 to 6GB of data to the on-board computer, depending on the chosen sensory configuration. With such large amounts of data to process, the computing equipment was designed with scal-

Figure 2.5: *Various mounting configurations:* **a) (left):** *dual stereo sensors mounted on top of vehicle* **b) (center):** *dual stereo sensors mounted on hood of vehicle,* **c) (right):** *experiment with an external sensor configuration*

ability as a guiding principle. For this purpose, A disk-less cluster arrangement was chosen essentially to provide the option of adding computing nodes as necessary. Currently, the on-board computer is composed of 16 computing nodes distributed over four boards networked with a gigabit switch. The nodes and the switch are contained inside a portable server case which in turn can be installed on the back seat or in the trunk of the vehicle. The computer and instrumentation are powered with a 1500W inverter connected directly to the battery of the vehicle. The instrumentation can be run continually without battery drainage. See Figure 2.6 for the schematics of the physical instrumentation.

2.6.2 Mounting Configurations

The visual sensors instrumenting the vehicle can be mounted in three distinct configurations. Figure 2.5 a) and c) depict an external, top-mounting of the dual stereo head, while Figure 2.5 b) shows an external hood-mounting configuration. Both of these set-ups do not hinder visibility for drivers. However, such external configurations limit the use of the instrumented vehicle to periods of clement weather (without rain, fog, or snow). To counter this limitation, the dual stereo head system was also designed to be mounted inside

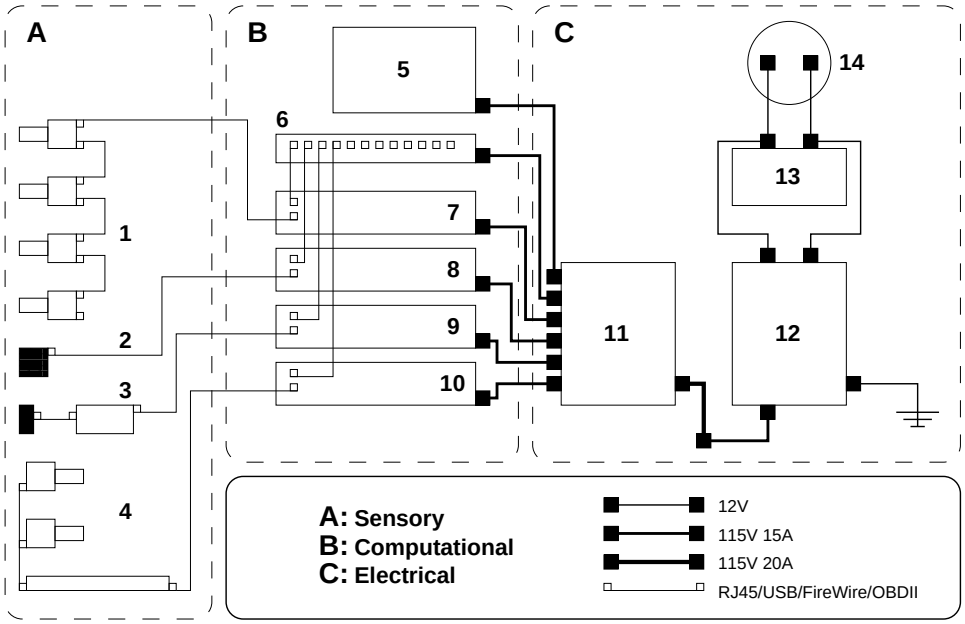

Figure 2.6: *Schematic vehicle instrumentation: 1): dual stereo sensors 2): GPS unit with USB interface 3): OBD-II to USB interface 4): FaceLAB 5*^{TM} *5): 19-inch LCD display 6): gigabit network switch 7): 3.0GHz quad-core master node with 128G Solid State Drive (SSD) 8), 9), 10): 3.0GHz quad-core slave nodes 11): 20A power conditioner 12): 1500W power inverter 13): vehicle battery 14): 140A vehicle alternator*

the front windshield of the vehicle (see Figure 2.4 b)). While this configuration allows the operation of the vehicle in variable weather conditions, it nonetheless hinders driver visibility substantially (a two-hour training session in closed-circuit is required before the vehicle can be safely driven on public roads). Another unintended effect created by this configuration is the visual distortion introduced by the presence of the windshield, which is not currently modeled by our calibration process, as it differs from radial and tangential distortion. Consequently, the quality and density of the raw stereo data is subjected to noticeable degradation (which lacks quantification at this time).

2.6.3 Software Services

The instrumented vehicle operational software architecture is based on a threaded publisher/subscriber model (see Figure 2.7). Each component executes on its own core, to ensure real-time performance. The RoadLAB recorder, depicted in Figure 2.8, receives images from the stereo heads at 30 fps, performs rectification, computes raw depth maps at frame rate, and saves the stereo images in a cyclic queue, which are to be written onto a solid state drive by an independent process which synchronizes with the recorder by means of semaphores.

The publisher/subscriber system receives information published by other software components such as the driver monitoring system[1], the OBD-II CAN-bus interface, and the GPS device. The recorder, in turn, may subscribe to various published elements and create instrumented sequences specifically designed for use in subsequent experiments. Alternatively, general-purpose instrumented sequences containing the totality of the published information can be produced. In general, the RoadLAB recorder may be used to provide

[1]FaceLAB 5^{TM}

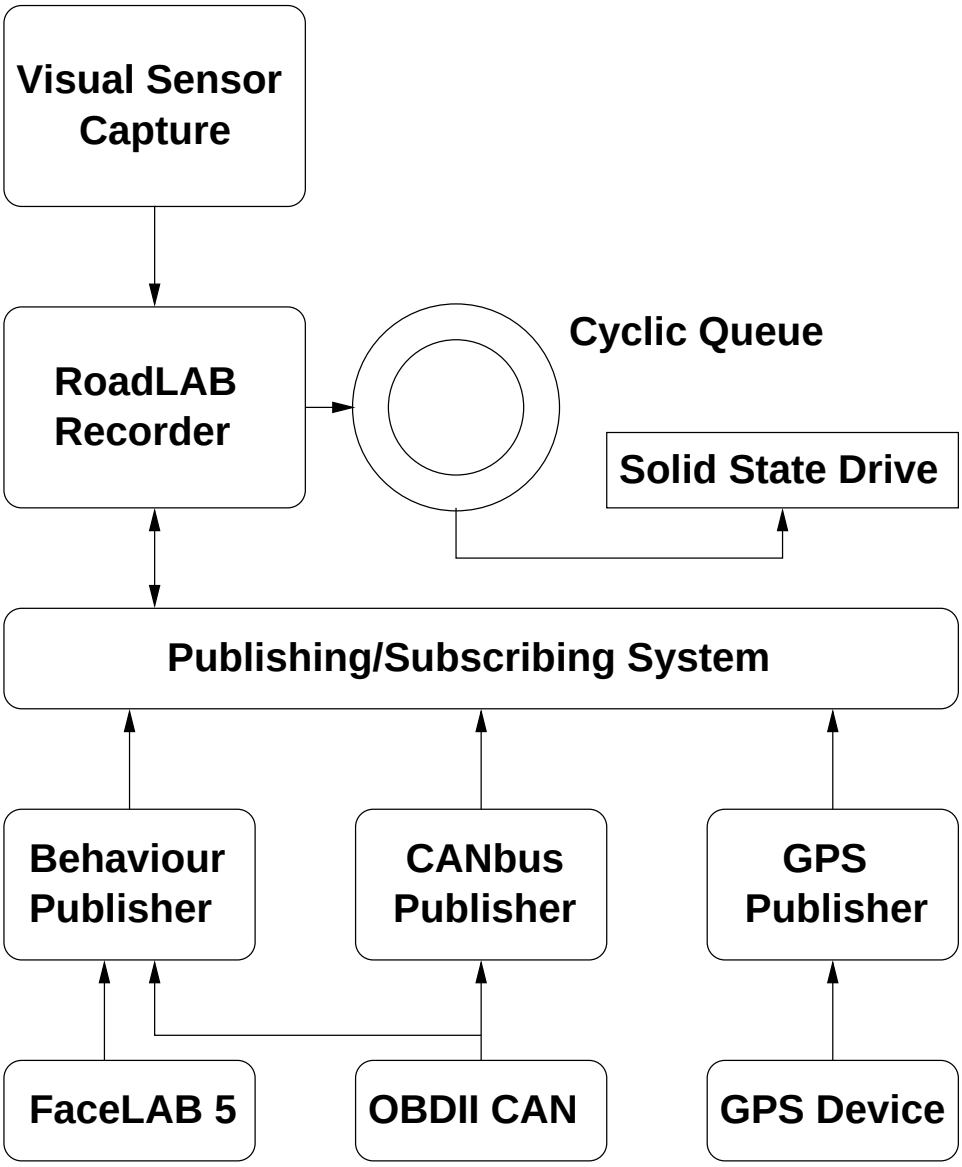

Figure 2.7: *Software services provided by the instrumented vehicle*

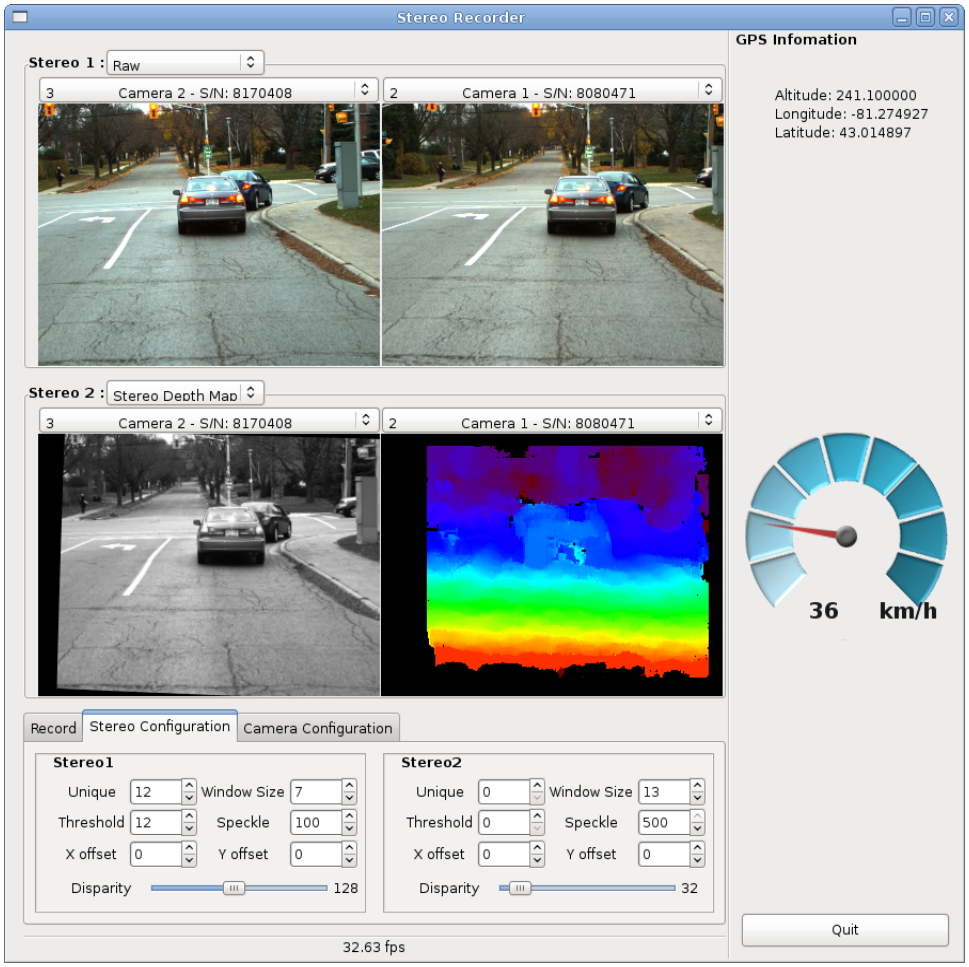

Figure 2.8: *The RoadLAB Sequence Recorder in operation inside the instrumented vehicle*

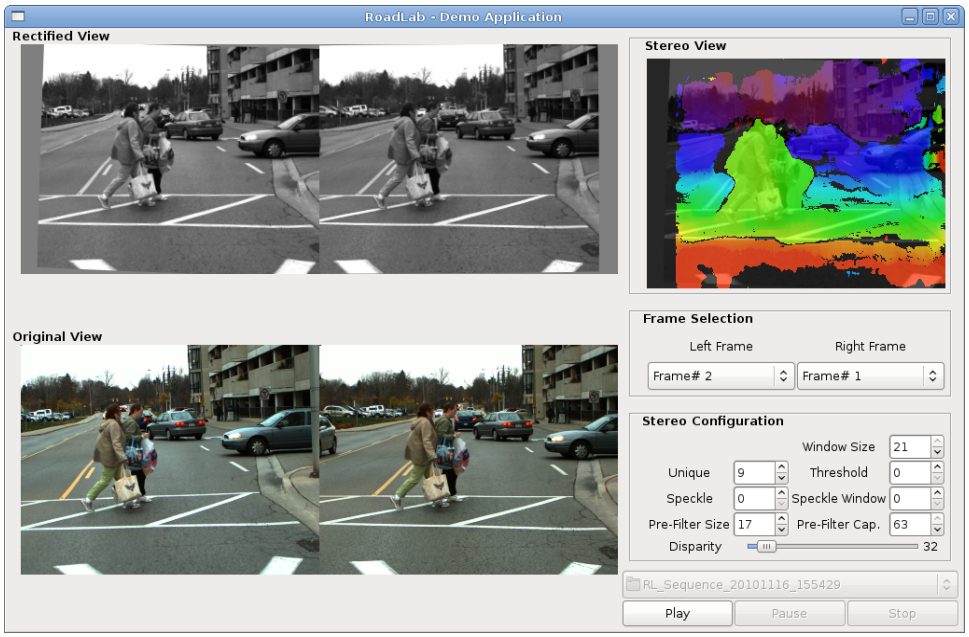

Figure 2.9: *A typical RoadLAB application using instrumented sequences produced with the vehicle operating in the recording mode*

real-time information to the resident i-ADAS application, or to produce instrumented sequences for in-laboratory experiments regarding the testing of sensing, integration, and i-ADAS algorithms.

2.6.4 Vehicular Operation

Operating the instrumented vehicle consists of several steps which must be carefully followed. First, the instrumentation must be installed in the vehicle. The computing nodes, gigabit switch, and the power conditioner (located inside the portable server case) are mounted onto the back seating area, while the stereo sensors are installed in the chosen configuration by way of vacuum devices. The GPS device magnetically attaches to the outside surface of the vehicle, and the CANbus to USB interface connects to the OBD-II outlet located under the vehicle instrumentation on the driver side. Once the sum of

Figure 2.10: *Real-time vehicle tracking experiment using the RoadLAB instrumented vehicle in i-ADAS mode*

these elements are connected to the on-board computer, the calibration process takes place. A large calibration plane (125 by 155 cm) is used to capture sets of calibration images (25 to 30) each with a different orientation of the plane. A distance from 8 to 15 m must be respected between the vehicle and the calibration plane in order to obtain accurate calibration parameters, depending on the lenses being used with the stereo systems. A minimum of three trained research assistants and 60 minutes are required for completing the instrumentation and the calibration processes. At this stage, the vehicle can be operated in the recording mode, the i-ADAS mode, or both. Figure 2.9 depicts a typical off-line RoadLAB application using instrumented sequences produced with the vehicle in recording mode. Figure 2.10 shows a real-time vehicle detection application, which is part of the resident i-ADAS software, and constitutes an example of the vehicle being operated in the i-ADAS mode.

2.6.5 Limitations

Several limitations are experienced while instrumenting a vehicle for purposes such as ours. On the sensing side, the use of vacuum devices to attach the instrumentation to the vehicle limits the time of continuous vehicular operation to 30 minutes. After such time, the vacuum device pumps must be operated once more, to securely maintain the equipment in place. In addition, long-range lenses (with long focal lengths), when installed on the stereo systems, are sensitive to vibrations generated by both the condition of the pavement and the operation of the vehicle, resulting in a degradation of the raw 3D depth data. This problem is made worse when the mounting configuration is located inside the windshield, as it introduces distortions that cannot be easily calibrated for. When conditions allow, an external mounting configuration

coupled with short to medium range camera lenses lead to noticeably improved 3D depth perception performance.

The availability of on-board computing power is inherently limited by the available space and electrical power in the vehicle. For instance, the use of high-resolution imagery would severely compromise our requirements for frame-rate processing. In this case, the problem may be addressed by replacing the computing nodes with GPUs, involving significant material costs. There is also the possibility of vehicle battery drainage with the use of high-end computing equipment, requiring the installation of a high-output, after-market vehicle alternator. In addition, our use of solid state drives limits the amount of time the vehicle can be operated in recording mode. In our case, this limit is between 10 to 30 minutes, depending on how many visual sensors are in use while recording.

While these limitations are significant, the use of the instrumented vehicle for the validation of our previously stated hypotheses is justified, as we proceed to demonstrate.

2.7 Methodological Considerations

Our current vehicular instrumentation is subservient to the validation of our hypotheses as described in this contribution and results from the following methodological considerations:

1. Instrumented sequences produced with test drivers are analyzed to determine what driving contexts correlate with cephalo-ocular behavior and to what extent this behavior can be considered predictive of driving actions. For this hypothesis to be tested correctly, drivers must

be in an adequate state of alertness, which is measured by both eye saccade frequency and fixation mean duration [26]. Subsequently, correlations between cephalo-ocular movements and resulting driving actions are measured. We hope to find out which cephalo-ocular behavior predict driver intentionality. Insights gained from this approach assist in the creation of effective predictive models of driver behavior.

2. Driver level of attention may or may not provide significance to observed cephalo-ocular behavior when various driving environments are factored in. From instrumented sequences, it is possible to measure correlations between attention (defined as frequency and mean duration of glances away from the roadway) and driving environments (urban, rural, highway, congestion), in order to infer the meaningfulness of cephalo-ocular behavior (excluding fatigue-related considerations). These results assist in determining what factors are descriptive of meaningful cephalo-ocular behavior as it relates to driving.

3. Correlation between increases in cognitive load, defined as degradation of mean reaction time, and density of information delivery using a variation of modalities (audio, tactile, and visual), defined as events per time unit, are measured in an attempt to evaluate the effects of warning systems on the cognitive loads of drivers.

4. Our last hypothesis relates to computer vision processes and our advances are evaluated against those that operate in similar contexts. In this case, metrics are standard and relate to performance, measured as computational efficiency and quantitative accuracy. In addition, protocols for V2V in terms of improving on-board sensory range and robust-

ness require other instrumented vehicles to communicate with, which are not available at this time, motivating our choice to explore this path with traffic simulators [5].

The in-vehicle laboratory as described in this contribution is capable of effecting the required measurements toward the validation of our hypotheses. Of particular importance is the extraction of driver behavior by using eye tracking and facial expression recognition techniques coupled with the maneuvers drivers apply to the vehicle, as obtained through the CANbus interface to form a basis for driver behavior prediction.

2.8 Performance Evaluation of Platform

The dual stereo systems constitute an essential component of the instrumented vehicle and for this reason, their performance (related to raw 3D depth data) is crucially important. We first consider the problem of range resolution, which is inversely related to object distance. The relationship governing range resolution is given by

$$\Delta r = \frac{r^2}{bf}\Delta d \tag{2.1}$$

where r is distance to object; f, focal length of imaging lens; b, stereo baseline length; and Δd, pixel size divided by the interpolation factor of the epipolar scan-line algorithm (for sub-pixel-precision 2D matching). The range resolutions for our dual stereo systems constitute a reliable indication of the error levels contained in the depth data, provided that calibration is accurate and that the depth measurements do not stem from incorrect 2D matches (due to occlusion, spatial aliasing, image noise, or related problems). Many dense stereo vision algorithms have been comparatively evaluated (including that of

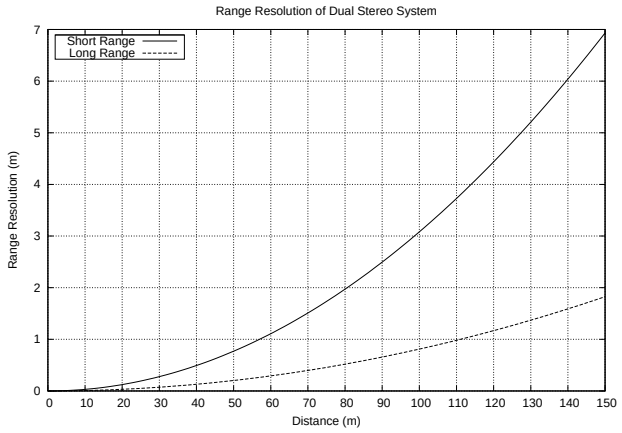

Figure 2.11: *Range resolution functions for dual stereo system, from* 0 *to* 150 m.

OpenCV, which we use) with image sequences for which true depth is available in terms of incorrect match density and resilience to noise [29]. The short-range stereo system has a baseline of length $b = 357$ mm, a smallest detectable 2D disparity of $\frac{1}{16}$ of a pixel, a focal length of $f = 12.5$ mm, and a physical pixel square size of 4.40 µm. The long range stereo system differs only in its baseline ($b = 678$ mm) and focal length ($f = 25.0$ mm). Figure 2.11 displays the range resolution functions for both stereo systems. As expected, the range resolution of the long range stereo pair surpasses that of the short range, due to an extended baseline and a longer focal length of the lens.

We have computed the average match density of both the long and short range stereo systems using instrumented sequences produced with the vehicle on public roads[1]. Results are reported in Table 2.1, where different values of the minimum disparity[2] were used. As can be observed, the short-range stereo

[1]The instrumented sequences used to perform these computations are publicly available at www.csd.uwo.ca/faculty/beau/roadlab_download/index.html.

[2]The minimum disparity parameter controls the offset to the disparity search window. Increasing positive values have an effect identical to augmenting the convergence of the stereo cameras.

Table 2.1: STEREO MATCH DENSITY FOR SHORT AND LONG RANGE SYSTEMS, WHERE d IS MINIMUM DISPARITY AND D IS MATCH DENSITY WITH STANDARD DEVIATION σ

Stereo Average Density							
Short Range				**Long Range**			
$d = 32$		$d = 64$		$d = 64$		$d = 96$	
D	σ	D	σ	D	σ	D	σ
71.6%	9.0%	82.5%	10.1%	49.4%	7.7%	41.3%	7.5%

system performs better in terms of density, due to several factors, including the reported fact that operational vibrations introduce more noise in long-range systems.

Each instrumentation layer as shown in Figure 2.1 has access to four cores (one node) to perform its real-time tasks. A total of sixteen cores are available for the four instrumentation layers. Currently, only one in the four available cores for each layer is in use. While the software is in the later stages of development, its current performance at 30Hz (for all layers) is consonant with the rate at which the visual sensors sample the environment. As the software modules are completed, the use of the remaining cores may become necessary to sustain the current performance. In the case where this would still be insufficient, an entire node can be added within the current configuration without any difficulty.

The performance of the quad-core computing nodes is largely sufficient to execute the stereo software at frame rate (30 fps). While one core suffices for the stereo computation, other cores may also be involved in processing other visual aspects of the captured frames and hence the speed at which frames can be transferred from one node to another is a critical constraint. By way of a high-end gigabit switch, the cores transfer frames (with resolution of 320

by 240 pixels) between nodes at 1.4MHz (or 0.7 ms per frame), a speed which does not impede on the performance of the system. Additionally, the highest transmission rate on the OBD II CANbus was measured at 200Hz, while our system reads and stores CANbus status at 2MHz, ensuring that no incoming message could be missed out[1].

2.9 Conclusion

We have addressed the problem of vehicle instrumentation as an experimental platform for the design of i-ADAS, while maintaining our requirements for physical portability and computational scalability. We framed the data processing strategy of the instrumentation within a layered approach in which data abstraction increases with the number of layers. The predictive behavioral model was also integrated with our layered structure, yielding a comprehensive implementation for hardware, software, and data abstraction framework. The resulting in-vehicle laboratory, its various configurations, software services, and operation modes were described in depth. We demonstrated that this platform, in spite of its limitations, can be effectively used to address the hypotheses we formulated in relation to the design of i-ADAS.

[1]Performance ratings of other aspects of our instrumentation such as the GPS device (GloablSat BU-353) and FaceLAB 5^{TM} are published by manufacturers and not reported herein.

Bibliography

[1] National Highway Traffic Safety Administration. Traffic safety facts 2006: A compilation of motor vehicle crash data from the fatality analysis reporting system and the general estimates system. Technical Report HS 810 818, US DOT, 2006.

[2] A. Amditis, K. Pagle, S. Joshi, and E. Bekiaris. Driver-vehicle-environment monitoring for on-board driver support systems: Lessons learned from design and implementation. *Applied Ergonomics*, 41:225–235, 2010.

[3] D.H. Ballard, M.M. Hayhoe, F. Li, and S.D. Whitehead. Hand-eye coordination during sequential tasks. *Philos. Trans. R. Soc. Lond. B*, 337:331–339, 1992.

[4] D.H. Ballard, M.M. Hayhoe, and J.P. Pelz. Memory representations in natural tasks. *J. Cog. Neurosci.*, 7:66–80, 1995.

[5] M.A. Bauer, K. Charbonneau, and S.S. Beauchemin. V2eye: Enhancement of automated visual perception from v2v. In *Consumer Communications and Networking Conference (CCNC)*, pages 737–738, Las Vegas, USA, January 2011.

[6] S.S. Beauchemin, M.A. Bauer, D. Laurendeau, T. Kowsari, J. Cho, M. Hunter, and O. McCarthy. Roadlab: An in-vehicle laboratory for developing cognitive cars. In *accepted, 23rd International Conference on Computer Applications in Industry and Engineering (CAINE)*, Las Vegas, USA, November 2010.

[7] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta. Artificial vision in road vehicles. *Proceedings of the IEEE*, 90(7):1258–1271, 2002.

[8] J.M. Collado, C. Hilario, A. de la Escalera, and J.M. Armingol. Self-calibration of an on-board stereo-vision system for driver assistance systems. In *IEEE Intelligent Vehicle Symposium, Tokyo, Japan*, pages 156–162, 2006.

[9] B. Donmez, L.N. Boyle, J.D. Lee, and G. Scott. Safety implications of providing real-time feedback to distracted drivers. *Accid. Anal. and Prev.*, 39(3):581–590, 2007.

[10] D.M. Gavrila, U. Franke, C. Wohler, and S. Gorzig. Real-time vision for intelligent vehicles. *Instrumentation and Measurement Magazine*, 4(2):22–27, 2002.

[11] D.M. Gavrila, M. Kunert, and U. Lages. A multi-sensor approach for the protection of vulnerable traffic participants - the protector project. In *IEEE Instrum. and Measurem. Tech. Conf.*, pages 2044–2048, Budabest, Hungary, May 2001.

[12] D. Jiang and L. Degrossi. Ieee 802.11p: Towards an international standard for wireless access in vehicular environments. In *IEEE Vehicular Technology Conference*, pages 2036–2040, 2008.

[13] K. Ker, I. Roberts, T. Collier, F. Beyer, F. Bunn, and C. Frost. Post-license driver education for the prevention of road traffic crashes; a systematic review of randomized controlled trials. *Accid. Anal. Prev.*, 37(2):305–313, 2005.

[14] K. Ker, I. Roberts, T. Collier, F. Beyer, F. Bunn, and C. Frost. Post-license driver education for the prevention of road traffic crashes; a systematic review of randomized controlled trials. *Accid. Anal. Prev.*, 37(2):305–313, 2005.

[15] E. Krug. Road traffic injuries. *WHO Overview Fact Sheet*, 2000.

[16] M.F. Land. Eye movements and the control of actions in everyday life. *Prog. in Ret. and Eye Res.*, 25:296–324, 2006.

[17] M.F. Land. Eye movements and the control of actions in everyday life. *Progress in Retinal and Eye Research*, 25(3):296–324, 2006.

[18] M.F. Land and J. Horwood. Which parts of the road guide steering? *Nature*, 377:339–340, 1995.

[19] J.D. Lee. 50 years of driving safety research. *Human Factors*, 50(3):521–528, 2008.

[20] Y. Lee, J.D. Lee, and L.N. Boyle. Visual attention in driving: The effects of cognitive loads and visual disruption. *Human Factors*, 49(4):721–733, 2007.

[21] J.F. Liu, Y.F. Su, M.K. Ko, and P.N. Yu. Development of a Vision-Based Driver Assistance System with Lane Departure Warning and Forward Collision Warning Functions. *Computing: Techniques and Applications*, pages 480–485, 2008.

[22] I. Masaki. Machine vision systems for intelligent transportation systems. *Intelligent Systems*, 13(6):24–30, 1998.

[23] L. Petersson, L. Fletcher, and A. Zelinsky. A framework for driver-in-the-loop driver assistance systems. *IEEE Intelligent Transportation Systems*, pages 771–776, 2005.

[24] K. Rayner. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, 124(3):372–422, 1998.

[25] M. Rockl, P. Robertson, K. Frank, T. Strang, and G.A. Center. An architecture for situation-aware driver assistance systems. In *IEEE 65th Vehicular Technology Conference*, volume 4, pages 2555–2559, 2007.

[26] D. Shinar. Looks are (almost) everything: Where drivers look to get information. *Human Factors*, 50(3):380–384, 2008.

[27] M.L. Sichitiu and M. Kihl. Inter-vehicle communication systems: a survey. *IEEE Communications Surveys & Tutorials*, 10(2):88–105, 2008.

[28] B.G Stoneman and D.J. Bowers. Development of an instrumented vehicle for traffic and safety assessmennt. In *7th Int. Conf. on Road Traffic Monitoring*, pages 115–118, London, England, April 1994.

[29] H. Sunyoto, W. van der Mark, and D.M. Gavrila. A comparative study of fast dense stereo vision algorithms. In *IEEE Intell. Vehicles Symp.*, pages 319–324, Parma, Italy, June 2004.

[30] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian,

and P. Mahoney. Stanley: The robot that won the darpa grand challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.

[31] Y. Umemura. Driver behavior and active safety. *R&D Review of Toyota CRDL - Special Issue*, 2004.

[32] Y. Umemura. Driver behavior and active safety. *R&D Review of Toyota CRDL - Special Issue*, 2004.

[33] W. van der Mark and D.M. Gavrila. Real-time dense stereo for intelligent vehicles. *IEEE Trans. Intell. Trans. Syst.*, 7(1):38–50, 2006.

[34] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

Chapter 3

Lane Detection System

This Chapter is a reformatted version of the following article:

T. Kowsari, S.S. Beauchemin, and M.A. Bauer, *Map-Based Lane and Drivable Area Detection*, submitted to, VISAPP 2014 Conference, Lisbon, portugal.

In this Chapter we propose a map based lane detection approach which can robustly detect the lanes in the city as well as rural roads and highways. In addition we present an algorithm for finding the obstacle free and safely drivable zone in the lanes based on the stereo depth map of the scene.

3.1 Introduction

Today, almost every new vehicle has some form of Advanced Driving Assistance System (ADAS). From adaptive cruise control, collision avoidance, and lane crossing warning systems to parking assistance, ADAS has made driving a safer and more enjoyable task. While a simple driving assistance system still requires a wealth of information on the state of the vehicle and its relationship to the immediate environment, intelligent ADAS requires even more, including

information on the state of the driver. Furthermore, the relative position and speed of other vehicles (and obstacles) constitute essential informational elements in the determination of lane-based safe and driveable areas directly located in front of the vehicle. In this contribution, we present an innovative lane detection system which combines GPS information and a global lane map with a forward facing vehicular stereo system to achieve robust lane detection. In addition, the stereo depth map enables the detection of of lane-based, obstacle-free areas.

Lane detection may appear trivial, at least in its basic setting. For instance, a relatively simple Hough transform-based algorithm can be used to detect the host lane for a short distance ahead without any tracking. This method proves effective in roughly 90% of the highway cases [2]. However, lane detection is considered a very challenging task when lanes other than the host one, obstacles of all kinds, and sharp turns are taken into account. Figure 3.1 shows a few examples of challenges posed by these conditions. The absence of lane markers (or worn-off ones), various lane shapes and sizes, occlusion, illumination changes, and weather conditions are among the reasons why lane detection is not as simple as it seems.

A recent lane and road boundary detection survey [5] explored a large body of research on lane detection and demonstrates the common work flow (at least partly) of lane detection systems (see Figure 3.2). Gradient-based feature detection [15, 12, 16], steerable filters [10], box filters [6, 19], or learning-based lane pattern recognition [4] are just a few examples of numerous feature detection approaches used in the literature.

Lane models, such as straight lines [9, 13, 14], parabolic curves [6, 10], semi-parametric formulations such as splines [9], or active contours [16] are found in the literature. Different model-fitting methods have been adopted

Figure 3.1: *Challenges of lane detection:* **a) (top-left):** *different curvature and shapes* **b) (bottom-left):** *vanished lane markers* **c) (top-right):** *splitting and merging lanes* **d) (bottom-right):** *occlusion and clutters*

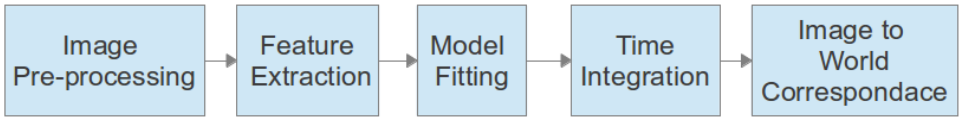

Figure 3.2: *Common work flow of a typical lane detection system.*

including RANSAC [16], particle swarms [20], energy-based optimization [16], genetic algorithms [15], and more. Despite this vast body of research, there are problems which yet remain to be satisfactorily addressed:

- Lane markings cannot be detected with range finders or other types of sensing that do not provide visible spectrum images. Even when sensors are adapted to lane marking detection, external problems arise, such as adverse weather, weak illumination, and worn-off markings, among others. Only a few authors in the literature have used specialized sensors such as line sensors [11] or GPS [7] to assist the detection process. In

this contribution we demonstrate how GPS and vehicle speed obtained from the internal network of the vehicle (CANbus) may be used in the design of a robust lane detection algorithm.

- Except in a few instances [9, 6], in almost the entire lane detection literature, lane models are not taking splitting and merging lanes (such as left turn lanes or opening and closing lanes) into account. Models often consist of parallel lanes without any distortion or starts and end to them. We have used a very simple yet flexible way of representing lanes such that all types of lanes can be represented and detected in most situations.

- Current lane detection algorithms are usually designed and tested either on highways or rural roads where sharp changes in lane position and orientation are not often observed. Our approach was tested successfully in dense urban areas where sharp turns, vehicle clutters, lane marker coverage, buildings, or other urban artifacts distract conventional lane detection algorithms.

- Most times, the most important lane from the point of view of the detection process is the host lane. However, in some cases we are interested in being able to describe a more complex environment such as the sum of lines adjacent to the host one.

We first provide a map-based framework which uses the GPS, vehicular speed, and a pre-loaded digital lane map as inputs to the lane detection algorithm. We then present the lane feature detection mechanism together with a particle swarm based tracking algorithm which fits the map with the lanes in the images. Subsequently, we use a simple yet effective stereo depth-based

obstacle detection by which we find the obstacle-free lane areas in front of the vehicle.

This contribution is organized as follows: Section 3.2 introduces the global lane map and lane modeling, Section 3.3 provides lane features and the Particle Swarm Optimization (PSO) algorithm, Section 3.4 describes the obstacle detection mechanism and the method to compute the obstacle-free lane areas, Section 3.5 presents the experimental results, and Section 3.6 offers a conclusion.

3.2 Lane Model

We present a global lane model for lane detection. While this type of model is not very common in the literature, we believe that it provides key advantages to the development of robust lane detection mechanisms. Using a lane map containing all lane paths and vehicle location on that map (with GPS or other methods for localization) facilitates the lane detection process and results in a more robust approach to the problem. To form the required lane maps, we annotated lanes in images provided by Google Earth satellite imagery.

In most of the methods found in the existing literature, it is generally assumed that the lane markers on the ground plane are approximately parallel. However, in reality, lane markers do not conform to this assumption. Even on roads where there is no splitting or merging of lanes, there are frequent lane shape distortions. In addition, most methods are concerned with the detection of the host lane only. We propose that modeling multiple lanes can significantly contribute to the robustness of lane detection algorithms, as any detectable part of a lane assists in preserving stability, especially in the absence of other cues. In light of this, it is believed that a robust model should have

the following properties:

- The model should address the *observed* shapes of lanes.
- In addition to the detection of the host lane, the model should be able to detect visible adjacent lanes.
- The model should include splitting and merging lanes (for instance, left turn lane parts in the center of the road at intersections or highway merging lanes)

The model contains a number of splines which model the entire map of the region of interest. Each spline is a lane marker and consists of points whose absolute positions on the map are their GPS latitude and longitude. In addition, these splines are binned into grid buckets representing non-overlapping contiguous regions each $500m^2$ in size. The sum of these buckets cover the entire lane map.

Each time the vehicle records data (it does so at 30Hz), a search for spline buckets that are most probably visible occurs, given the vehicle's position and orientation, and the front stereo system viewing angle. The lane marking splines from the selected buckets are subsequently sorted in space with respect to the perpendicular of the direction of the vehicle, which amounts to a sorting from left to right in terms of visibility from the point of view of the stereo system.

With t sorted lane marking splines hypothetically forming $t-1$ lanes and two out-of-road areas, and the position (latitude and longitude) and orientation (obtained with the vector formed from the last two GPS coordinates) of the vehicle, the positions of the splines are converted into the reference frame of the front stereo system (with its origin at the optical center of the left cam-

Figure 3.3: *Images from the map building application* **a) (left):** *visible lanes in presence of traffic* **b) (center):** *Splitting lane sample* **c) (right):** *Several neighboring lanes*

era) in meter units. Each lane $L_i \in \{L_0, \ldots, L_{t-1}\}$ is composed of two lane marking splines.

In order to specify the modalities of splitting and merging lanes, the model requires the opening and closing distances of the lanes from the vehicle. To address this, at each time interval, we assign $t-1$ variables LaneCloses(i) for the closing distance of each lane and another $t-1$ variables LaneOpens(i) with the same size for the opening distance of each lane. The opening distances for the lanes which are already open are set to 0, while the closing distances for the lanes that are not yet closed are set to ∞.

Since we require our model to detect obstacle-free areas in the lanes, we considered another $t+1$ variables LaneBlocks(i) which contain either ∞ to signify not blocked or a distance in meters indicating that there is an obstacle in this lane at that distance.

3.2.1 Spline Lane Marker Model

We adopted the Catmull-Rom spline formalism for the lane-marking splines [3] since it interpolates the control points. For each spline segment between control points P_i and P_{i+1}, the spline is obtained with control points P_{i-1} to P_{i+2} as [18]:

$$S(t) = \begin{bmatrix} 1 & t & t^2 & t^3 \end{bmatrix} M \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix} \tag{3.1}$$

where $S(t)$ is either the x or y element of the coordinates of the curve points, $t \in [0, \ldots, 1]$ and

$$M = \frac{1}{2} \begin{bmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & \frac{1}{2} & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

3.2.2 Generating the Lane Map

Google Earth satellite images are used to build the lane maps. Satellite images adequately fit our purposes as lane markers are not occluded by vehicles or other urban structures. These images can also be addressed directly by longitude and latitude which is desirable since we use GPS coordinates to locate the vehicle on the map and extract hypothetically visible lanes from the stereo images. We created an application which uses Google static API to obtain and display bird's eye images of the region of interest at requested positions. (see Figure 3.3). This application also allows a user to draw and edit splines as

lane markers. The user is also able to navigate through the map and follow the road while drawing lanes. The resulting data is saved as a set of lane-marking splines, each of them containing a set of control points. In our experiments, we extracted a path that was traveled by the experimental vehicle within the city of London, Ontario. This path consists of 94 lanes and lane segments, including right and left turn lanes.

3.3 Model Fitting Using a Particle Filter

With the knowledge of the position and orientation of the vehicle within the lane map, we proceed to fit our lane model onto the detected lane features in the left stereo image.

Since the GPS data frequency (1Hz) is significantly slower than that of the front stereo system (30Hz), the most recent speed data of the vehicle obtained from the CANBus is used to extrapolate the most recent available GPS data to coincide with the most recent image frame from the front stereo system. This can be thought as a form of synchronization of the GPS device and the front stereo system. In addition, the GPS data has a relatively large error (we observed a $\pm$5m error), and can be used only as a seed for the lane fitting process.

With the approximate position and orientation of the vehicle, the visible parts of the lane map in the image can be identified. The lane-marking splines are projected onto the stereo left image and an optimization algorithm attempts to find the best relative change in the position and orientation of the vehicle which best fits the projection with the lane features in the image. This optimization yields two parameters δX and $\delta\theta$ which correct the current vehicle position and orientation obtained form the GPS at each frame.

In order to project the lane markers onto the image we need to know the ground plane equation parameters in the camera coordinate system. Even though the ground plane parameters are very stable, we noticed that including a correction parameter $\delta\lambda$ representing an offset to the pitch angle of the ground plane improves the accuracy of the projection process by compensating for the unexpected tilt variations due to vehicle suspension.

3.3.1 Ground Plane Estimation

The ground parameters needed for projecting the lanes on the image can be computed from the depth map obtained from the stereo system. With rectified stereo images, finding disparities and hence depth map merely consists of a 1-D search with a block matching algorithm (our implementation uses the stereo routines from Version 2.4 of OpenCV) Assuming that the ground plane equation is of the form

$$ax + by + cz = d \tag{3.2}$$

where $\vec{n} = (a, b, c)$ is the unit normal vector to the plane, we pose

$$d = \frac{1}{\sqrt{a'^2 + b'^2 + c'^2}} \tag{3.3}$$

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = d \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix} \tag{3.4}$$

With the coordinates of 3D points in the reference system of the left camera

$$(X_i, Y_i, Z_i) \tag{3.5}$$

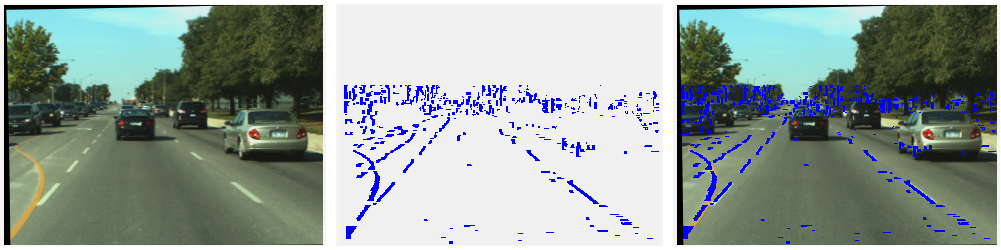

Figure 3.4: **a) (left):** *The raw image* **b) (center):** *Low-level lane feature detection result* **c) (right):** *Features depicted on the image.*

we can write

$$\mathbf{A}\mathbf{x} = \mathbf{B} \tag{3.6}$$

and solve for $\mathbf{x}$ in the least-squares sense as

$$\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{B} \tag{3.7}$$

where

$$\mathbf{A} = \begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix}$$

Often times the ground surface leads to inordinate amounts of outliers, due in part to a lack of texture from the pavement or other driveable surfaces. With the sensitivity of least-squares to outliers being known, we resort to the use of RANSAC in selecting the inliers and obtain a robust estimation of the ground plane coefficients, in the following way:

1. randomly select three points from the 3D points believed to be representative of the ground plane

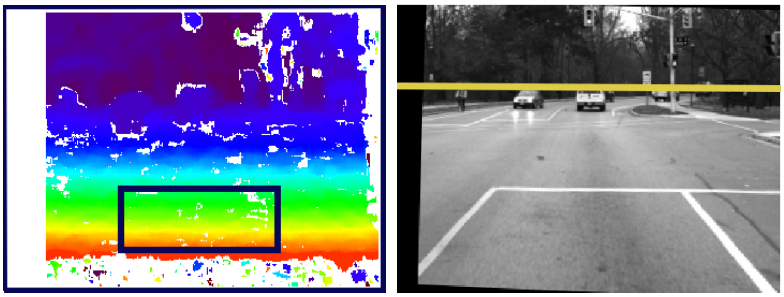

Figure 3.5: **a) (left)**: *Color-coded stereo depth map and the region used for ground plane estimation* **b) (right)**: *Horizon line estimated from ground plane*

2. compute the coefficients of the plane defined by the randomly selected points using (3.6)

3. count the points whose distance to the plane is less than a threshold ϵ

4. repeat these steps n times where n is sufficiently large[1]

5. among the n fits choose the largest inlier set which respect to ϵ and compute the coefficients of the ground plane this time using least-squares as in (3.7)

The plane parameters are averaged over a short period of time in order to stabilize them further. The coefficients of the plane are recomputed at each new stereo frame arrival. However, in cases when the number of depth values is low (poor texture, etc.) or other vision modules indicate the presence of a near obstacle, the coefficients of the ground plane are not recomputed, the previous parameters are used instead.

The horizon line is approximated by intersecting the plane parallel to that of the ground and passing through the focal point with the image plane of the sensor, and converting to image coordinates using calibration parameters.

[1]Choosing $n > 20$ does not significantly improve the number of inliers with respect to ϵ.

Figure 3.5 displays a sample disparity map and estimated horizon line, used in our hypothesis generation mechanism.

Introducing the tilt parameter $\delta\lambda$, the ground plane equation becomes:

$$ax + by + (c + \delta\lambda)z = d \tag{3.8}$$

3.3.2 Likelihood Function

The estimation of the best fit parameters between projected lane-marking splines and the detected lane features in the left stereo image is performed by defining a likelihood function

$$\mathcal{L}(z|\mathbf{x}) \tag{3.9}$$

where z is a particular parameter fit, and $\mathbf{x} = (\delta x, \delta\theta, \delta\lambda)$. Estimating this likelihood function requires first the detection of lane boundary features from the stereo imagery. Image features must satisfy a number of constraints before they can be considered as lane boundary features, such as being located on the ground plane, featuring a lighter gray level than that of the ground plane, and be contained within two significant gradient values of a predefined width (which depends on the observed depth).

The algorithm to detect lane boundary features is formally described in Algorithm 1 and uses the left camera stereo image I and its depth map I_d as inputs to produce a Gaussian smoothed lane boundary feature image F, such as that displayed in Figure 3.4b. Constants found in the algorithm are α and β, used for computing the width expectation of the lane markings $L_{\max}$, factored by their distance from the vehicle. Constants NL and LD indicate the state of the lane edge search. NL represents the state in which no lanes are detected, while LD is its complement. Threshold τ_h represents the minimum gradient

value required for a transition from NL to LD. Constant O_h is the minimum variation in height from the ground plane for a pixel to be considered part of an obstacle. O_h and τ_h depend on imagery and are experimentally determined.

Algorithm 1 *Lane Feature Detection Algorithm*

```
G ← 1D Gaussian row smoothing of I with σ = 0.5
G ← horizontal gradient of G using 3-point central differences
Remove the values corresponding to obstacles from G using threshold O_h
State ← NL
F initialized to 0
for all rows i in I starting from the image bottom do
  L_max ← β − iα
  Count ← 0
  for all column j in I do
    if (G_{i,j} > τ_h ∧ (State = NL ∨ Count > L_max)) then
      State ← LD
    end if
    if (State = LD) ∧ (G_{i,j} < −τ_h) then
      for k = j − Count → j do
        F_{i,k} ← 1
      end for
      State ← NL
      Count ← 0
    end if
  end for
end for
F ← 1D Gaussian row smoothing of I with σ = 0.5
```

The likelihood function (3.9) may be estimated using the extracted lane marking features F and the sorted (from left to right) lane marking splines contained in the visible spline buckets. The lane-marking splines from the map are aligned with the direction of the vehicle by a rotation and then projected on the image plane so as to find a best fit with the detected lane marking features. Assuming that the Z axis of the 3D reference frame of the front stereo system of the vehicle makes an angle θ with the Y axis of the 2D reference frame of

the lane map, a spline point $\mathbf{Q} = (X, Y)$ in the coordinates of the lane map is rotated according to:

$$\begin{bmatrix} X_r \\ Z_r \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos(\theta) & -\sin(\theta) \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{3.10}$$

With the ground plane equation, we estimate the tilt-corrected height coordinate in the reference frame of the stereo system as:

$$Y_r = \frac{d - aX_r - Z_r(c + \delta\lambda)}{b} \tag{3.11}$$

where $\mathbf{Q}_r = (X_r, Y_r, Z_r)$ is the 3D spline point expressed in the reference frame of the stereo system. The projection of $\mathbf{Q}_r$ onto the stereo imaging plane is performed by applying the classical projection matrix P obtained for the calibration process of the stereo system:

$$w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = P \begin{bmatrix} X_r \\ Y_r \\ Z_r \\ 1 \end{bmatrix} \tag{3.12}$$

where w is a scaling factor due to the use of homogeneous coordinates.

With the lane feature image F and the projected, visible lane-marking splines, the likelihood function becomes

$$\mathcal{L}(z|\mathbf{x}) = \sum_{(i,j)\in\mathbf{S}} F(i, j) \tag{3.13}$$

where $\mathbf{S}$ is the set of all projected points of the lane-marking splines.

3.3.3 Particle Filtering

With the likelihood function, we need to estimate the parameters $\mathbf{x}$ of the fit as:

$$x = \text{argmax}\mathcal{L}(z|\mathbf{x})_{\mathbf{x}} \tag{3.14}$$

Solving this optimization problem is not easily achievable by regular hill-climbing methods due to the non-concavity of the function. Since the search space is large, an exhaustive search is prohibitively expensive while the probability of finding the global maximum remains low [17].

A particle swarm method may be more appropriate. The particle swarm lane detection algorithm by Zhou [20] is a single image frame method, which we adapt here as a particle filter working on a sequence of frames[1]. Our approach consists of generating a set of uniformly distributed particles, each representing a set of possible values for parameters $\mathbf{x} = \delta x, \delta\theta, \delta\lambda$. The likelihood of each particle is estimated with (3.14).

At each iteration, each particle is replaced with a number of newly generated, Gaussian position-disturbed particles. The number of generated particles is proportional to the likelihood of the particle they replace. Their likelihood are estimated again with (3.14) and normalized. This ensures that the stronger particles generate more particles in their vicinity than the weaker ones. Particles with normalized likelihoods lower than a certain threshold are removed and, if the number of particles becomes less than a threshold, the process repeats.

These iterations eventually lead to groups of particles concentrated at the most likely answers in the search space and the particle with the maximum

[1]PSO is a population-based stochastic optimization method first proposed by Eberhart and Kennedy [8].

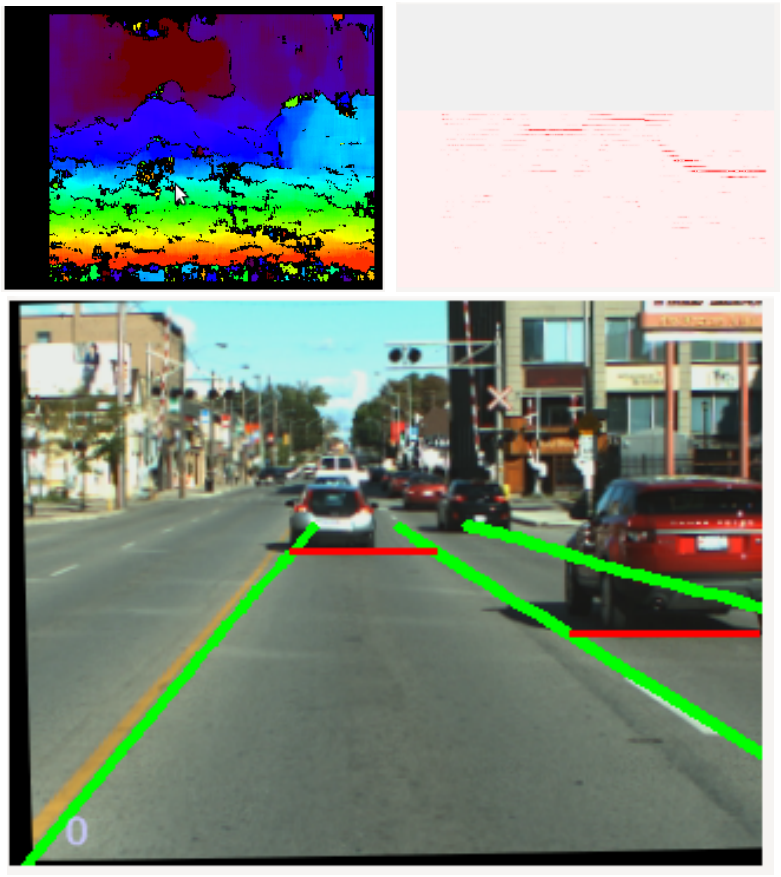

Figure 3.6: **a)(top-left):** *Color-mapped stereo depth map* **b) (top-right):** *Accumulated projected obstacle points* **c) (bottom):** *Results of obstacle-free areas detection*

likelihood is chosen as the solution. In addition, keeping the particles over time makes the particle filter to act as a tracker for the lane detection mechanism.

3.4 Obstacle Detection

With a set of detected lanes represented by projected splines, the stereo depth map can be used to locate obstacles within each detected lane. The inputs to the obstacle-free detection algorithm are the stereo disparity map I_d, the

classical projection and re-projection matrices P and D, the ground plane parameters a, b, c, d, and $\delta\lambda$, and the projected lane-marking splines. The output consists of the distance from the vehicle to first obstacle (if present) for each lane. The algorithm uses constant O_h as previously defined, and threshold O_t which is the minimum ratio of obstacle pixels to all pixels across a lane, for each row in the image.

The first stage of the algorithm consists of detecting pixels whose 3D positions computed as:

$$W \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = D \begin{bmatrix} u \\ v \\ d \\ 1 \end{bmatrix} \tag{3.15}$$

are not lying on the ground plane. The distance of the 3D point from the ground plane is obtained as:

$$\text{Dist} = aX + bY + (c + \delta\lambda)Z - d \tag{3.16}$$

The algorithm keeps an obstacle map O the size of the original image. The 3D coordinates of each pixel whose height from the ground plane qualifies it as an obstacle is projected onto the ground plane by setting its Y coordinate according to (3.11), and then projected onto the obstacle map O, using

$$w \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = P \begin{bmatrix} X \\ Y_g \\ Z \\ 1 \end{bmatrix} \tag{3.17}$$

where the corresponding image location in O is incremented by one.

The last stage of the algorithm consists of scanning all rows of image O from the bottom. In each row, between the boundaries of each lane which is not yet blocked, the values of O at the positions across the lane are summed up and divided by the total number of pixels in that lane, forming a lane ratio γ. If this ratio exceeds threshold O_t, the lane is assumed to be blocked by an obstacle at that row and the distance of the obstacle is recorded for that lane. The formal description of this algorithm is given in 2. Figure 3.6 shows an example of accumulated projected obstacle points map and the resulting free zone detection for the lanes.

Algorithm 2 *Obstacle-Free Zone Detection Algorithm*

O initialized to 0
for all $O(u, v)$ **do**
  Compute 3D coordinates of the point in the stereo reference frame using I_d and (3.15)
  Dist $\leftarrow aX + bY + (c + \delta\lambda)Z - d$
  if Dist $> O_t$ **then**
    $Y_g \leftarrow (d - aX - (c + \delta\lambda)Z)/b$
    Compute (u', v') using (3.17)
    $O_{(u',v')} \leftarrow O_{(u',v')} + 1$
  end if
end for
for all rows i in O **do**
  for all lanes **do**
    if lane ratio $\gamma > O_t$ and lane still open **then**
      Output the lane as a blocked lane at corresponding distance
    end if
  end for
end for

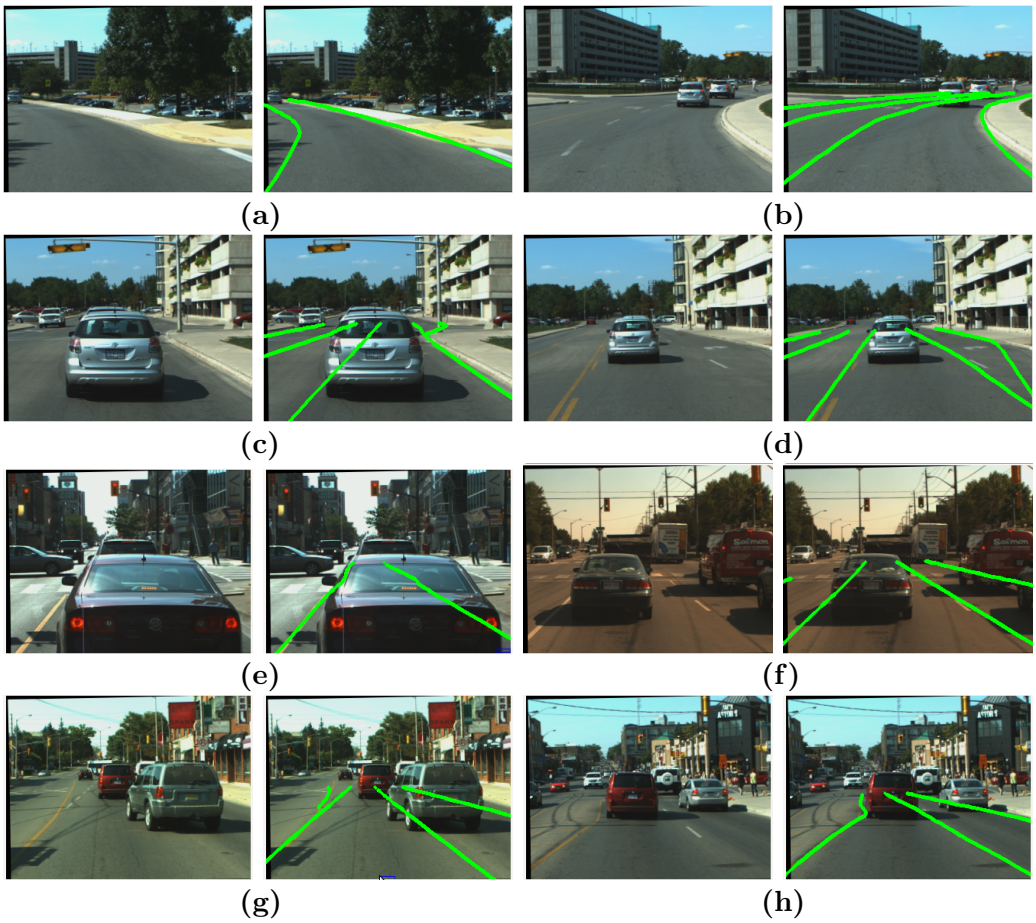

Figure 3.7: *Examples of lane detection results* **a)** *Out-of-sight lane markers* **b)** *Multiple curved lanes* **c)** *Occluded lane markers* **d)** *Splitting lane* **e)** *Urban distractions* **f)** *Multiple lanes partially occluded* **g)** *Splitting lane and occlusion* **h)** *Irregular lane shape*

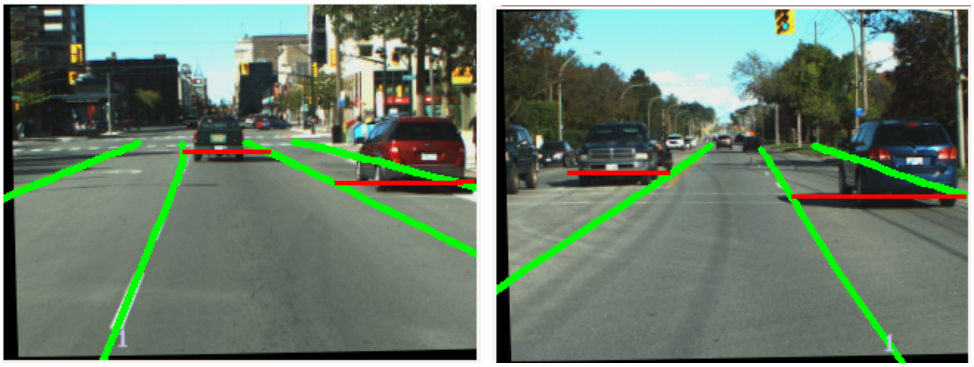

Figure 3.8: *Examples of obstacle-free area detection results* **a) (left):** *Ongoing traffic, within the detected lanes* **b) (right):** *Incoming traffic outside of detected lanes*

3.5 Experimental Results

We applied this approach to a set of sequences recorded form an instrumented experimental vehicle [1]. The implementation of the technique executes at 15Hz, including the stereo depth computation, ground plane detection, particle filtering for lane detection, and obstacle-free area estimation. Thirty initial particles for the particle swarm were used, and the stereo image size was 320 by 240 pixels.

Figure 3.7 shows samples of detected lanes and Figure 3.8 shows samples of detected blocking of the lanes by vehicles or other obstacles. The experiments subjectively demonstrated that the algorithm is robust to occlusion and partially worn-off or occluded lane markers and various urban artifacts. As observed, our technique remains stable, even for some frames without any evidence of lane markers, which is very difficult for most of the existing lane detection approaches. Even in the presence of significant lane marker occlusions, our approach still properly detects lanes.

One may argue that the requirement for GPS-addressable, lane-annotated

maps limits the areas in which this approach may be used, which is correct. However, we believe this approach can be used in most driving situations, so long as lane-annotated maps are automatically generated and made available to instrumented vehicles. Additionally, the confidence measure obtained from thresholding the likelihood function may be used to assess the reliability of detected lanes.

3.6 Conclusion

We proposed a map-based lane detection and obstacle-free area detection using lane-annotated maps, particle filtering, and stereo depth maps. Our main contribution consists of our lane model obtained from lane-annotated maps, allowing us to represent irregular, opening, and closing lanes that are often ignored in the current literature. Ironically, these types of lanes are crucially important for iADAS as they occur in critical areas such as intersections and merging and turning areas which constitute perilous zones. Our approach uses a robust model that does not entirely depend on an on-board imaging system easily lead astray by the presence of occluding obstacles and worn-off lane markers.

Bibliography

[1] S.S Beauchemin, M.A Bauer, D Laurendeau, T Kowsari, J Cho, M Hunter, and O McCarthy. Roadlab: An in-vehicle laboratory for developing cognitive cars. In *23rd International Conference on Computer Applications in Industry and Engineering (CAINE 10)*, pages 7–12. IEEE, 2010.

[2] Amol Borkar, Monson Hayes, Mark T Smith, and Sharathchandra Pankanti. A layered approach to robust lane detection at night. In *Computational Intelligence in Vehicles and Vehicular Systems, 2009. CIVVS'09. IEEE Workshop on*, pages 51–57. IEEE, 2009.

[3] Edwin Catmull and Raphael Rom. A class of local interpolating splines. *Computer Aided Geometric Design, Academic Press, New York*, page 317326, 1974.

[4] Hsu-Yung Cheng, Bor-Shenn Jeng, Pei-Ting Tseng, and K-C Fan. Lane detection with moving vehicles in the traffic scenes. *Intelligent Transportation Systems, IEEE Transactions on*, 7(4):571–582, 2006.

[5] Aharon Bar Hillel, Ronen Lerner, Dan Levi, and Guy Raz. Recent progress in road and lane detection: a survey. *Machine Vision and Applications*, pages 1–19, 2012.

[6] Albert S Huang, David Moore, Matthew Antone, Edwin Olson, and Seth Teller. Finding multiple lanes in urban road networks with vision and lidar. *Autonomous Robots*, 26(2-3):103–122, 2009.

[7] Yan Jiang, Feng Gao, and Guoyan Xu. Computer vision-based multiple-lane detection on straight road and in a curve. In *Image Analysis and*

Signal Processing (IASP), 2010 International Conference on, pages 114–117. IEEE, 2010.

[8] James Kennedy and Russell Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948. IEEE, 1995.

[9] ZuWhan Kim. Robust lane detection and tracking in challenging scenarios. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1):16–26, 2008.

[10] Joel C McCall and Mohan M Trivedi. Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. *Intelligent Transportation Systems, IEEE Transactions on*, 7(1):20–37, 2006.

[11] Yusuke Narita, Shunji Katahara, and Masayoshi Aoki. Lateral position detection using side looking line sensor cameras. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 271–275. IEEE, 2003.

[12] Marcos Nieto, Luis Salgado, Fernando Jaureguizar, and Jon Arróspide. Robust multiple lane road modeling based on perspective analysis. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 2396–2399. IEEE, 2008.

[13] Dean Pomerleau. Ralph: Rapidly adapting lateral position handler. In *Intelligent Vehicles' 95 Symposium., Proceedings of the*, pages 506–511. IEEE, 1995.

[14] Christopher Rasmussen and Thommen Korah. On-vehicle and aerial texture analysis for vision-based desert road following. In *Computer Vi-*

sion and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on, pages 66–66. IEEE, 2005.

[15] F Samadzadegan, A Sarafraz, and M Tabibi. Automatic lane detection in image sequences for vision-based navigation purposes. In *Proceedings of the ISPRS Commission V Symposium'Image Engineering and Vision Metrology*, 2006.

[16] Hiroaki Sawano and Minoru Okada. A road extraction method by an active contour model with inertia and differential features. *IEICE transactions on information and systems*, 89(7):2257–2267, 2006.

[17] E-G Talbi and Traian Muntean. Hill-climbing, simulated annealing and genetic algorithms: a comparative study and application to the mapping problem. In *System Sciences, 1993, Proceeding of the Twenty-Sixth Hawaii International Conference on*, volume 2, pages 565–573. IEEE, 1993.

[18] Alan Watt and Mark Watt. Advanced rendering and animation techniques: Theory and practice. *Reading, MA*, 1991.

[19] Shinq-Jen Wu, Hsin-Han Chiang, Jau-Woei Perng, Chao-Jung Chen, Bing-Fei Wu, and Tsu-Tian Lee. The heterogeneous systems integration design and implementation for lane keeping on a vehicle. *Intelligent Transportation Systems, IEEE Transactions on*, 9(2):246–263, 2008.

[20] Yong Zhou, Xiaofeng Hu, and Qingtai Ye. A robust lane detection approach based on map estimate and particle swarm optimization. In *Computational Intelligence and Security*, pages 804–811. Springer, 2005.

Chapter 4

Vehicle Detection System

This Chapter is a reformatted version of the following article:

T. Kowsari, S.S. Beauchemin, and J. Cho, *Real-Time Vehicle Detection and Tracking Using Stereo Vision and Multi-View AdaBoost", IEEE Intelligent Transportation Systems Conference*, Washington DC, USA, pp. 1255-1260, October 5-7, 2011.

In this Chapter, we propose a multi-layer, real-time vehicle detection and tracking system using stereo vision, multi-view AdaBoost detectors, and optical flow. By adopting a ground plane estimate extracted from stereo information, we generate a sparse set of hypotheses and apply trained AdaBoost classifiers in addition to fast disparity histogramming, for Hypothesis Verification (HV) purposes. Our tracking system employs one Kalman filter per detected vehicle and motion vectors from optical flow, as a means to increase its robustness.

4.1 Introduction

Most Driving Assistance Systems (DAS) rely on the detection of relevant features in the immediate environment of the vehicle, such as other vehicles, pedestrians, lanes, traffic signs, and other potential obstacles [2]. Many driving assistance systems such as adaptive cruise control, collision warning, blind spot monitoring, and park assist rely on some form of obstacle detection mechanisms. While obstacle detection in general can be made simpler with the use of active sensors, the resulting cross-talk and noise from other vehicles can potentially deteriorate the robustness of such systems. Moreover, for large distances and increased resolution, passive sensing seems more applicable [7].

Recently, visible spectrum vision-based vehicle detection has attracted a lot of attention due to improved machine vision algorithms and the availability of low-cost high computational power [33]. Real-time detection is vital for DAS as road-based imagery is highly dynamic (a vehicle moving at a speed of 120 kmh changes its position by 33 m every second). While very dependable techniques and methods for object detection have appeared [20], most of them require high-cost, advanced hardware to be executed in real-time, if at all possible.

This contribution proposes a real-time, multi-layer vehicle detection system using stereo vision, optical flow, and a machine-learning classifier, and is structured as follows: Section 4.2 reviews the recent vehicle detection literature, Section 4.3 presents our main approach, while Sections 4.4, 4.5, and 4.6 describe Hypothesis Generation (HG), Hypothesis Verification (HV), and Tracking respectively. Section 4.7 presents our experimental results while Section 4.8 offers a conclusion.

4.2 Related Literature

There exists a large body of research from the last two decades on DAS. With the goal of creating autonomous vehicles, many research groups have launched several projects in different aspects of DAS [37, 27, 6, 8, 36, 21, 35, 33].

Our focus is on detecting and tracking vehicles as imaged by passive sensors inside an instrumented vehicle. The vehicle detection stage is usually broken into three parts: Hypothesis Generation (HG), Hypothesis Verification (HV), and tracking. For HG, several contributions have proposed various methods, including the selective detection of vertical and horizontal edges [34], symmetry maps [4], color [9], stereo depth [23, 25, 40], texture [17], and motion [14], either in isolation or in combination.

In Region Of Interest (ROI) based HG, Cheng *et al.* use vanishing points obtained from the intersection of detected lanes [11] and achieved 20fps on conventional hardware. While lane-based vanishing point detection requires salient road markings, Sappa *et al.* computed ROIs with horizon lines obtained from stereo depth data [29]. In addition, Keller *et al.* proposed a similar algorithm but employ a B-Spline model for the road rather than a flat plane [18]. In both cases however, a lack of knowledge about obstacles on the road often leads to erroneous ground plane estimations. Conversely, samples of disparity-based obstacle detection methods are that of Jung *et al.* [16] and Mandelbaum *et al.* [24] in which the authors used disparity histogram peaks as evidence of obstacles.

The HV stage is often performed with either block matching [15, 28] or appearance-based methods, which take features such as Scale Invariant Feature Transform (SIFT) [22], Principal Component Analysis (PCA) [39], summation of intensity or Gabor filtering [31], to train a classifier into verifying generated

hypotheses. Different classifiers such as Support Vector Machines (SVMs) [31], neural networks [26], AdaBoost [38], and nearest neighbor [39] are often used in the literature.

4.3 Description of Approach

We propose a three-stage vehicle detection method which includes: Hypothesis Generation based on ground plane estimation, Hypothesis Verification with Haar-like features, an AdaBoost classifier, and disparity histogramming, followed by vehicle tracking using optical flow and Kalman filters.

With the depth map we estimate the parameters of a plane which fits the data in a near rectangle at the bottom of the image where there is no visible obstacle. The absence of obstacles in this region is determined by the absence of peaks within the part of the disparity histogram corresponding to the ground plane near the vehicle. By using RANdom SAmple Consensus (RANSAC) in estimating the parameters of the ground plane, the effects of depth outliers are minimized. The horizon line can in turn be estimated by intersecting the plane parallel to the ground plane and containing the focal point with the imaging plane of one of the sensors. The horizon line constitutes the basis for generating hypotheses. These are then tested with a set of AdaBoost classifiers trained with four different views of various vehicles from existing data sets. An integral disparity histogram is used to increase the robustness of the verification stage. Following these steps, optical flow data provides assistance in tracking sparse features on verified hypotheses. If the number of overlapping rectangles of similar scale containing detected vehicles exceeds a certain limit in a number of consecutive frames, they are then merged into a single rectangle and a Kalman filter is created to track the detected vehicle(s)

within the extent of the merged rectangle. The algorithm is discussed with greater detail in the next Sections.

4.4 Hypothesis Generation

With rectified stereo images, finding disparities merely consists of a 1-D search with a block matching algorithm (our implementation uses the stereo routines from Version 2.2 of OpenCV). As expected, the presence of an obstacle in the image creates a peak in the disparity histogram. Considering this, an obstacle-free area from the bottom portion of the image may be determined and then used to estimate the position of the ground plane.

Using the disparities and the calibration parameters of the stereo system, the 3D positions of the image pixels in the camera coordinate system are computed. Assuming that the ground plane equation is of the form

$$ax + by + cz = d \tag{4.1}$$

where $\vec{n} = (a, b, c)$ is the unit normal vector to the plane, we pose

$$d = \frac{1}{\sqrt{a'^2 + b'^2 + c'^2}} \tag{4.2}$$

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = d \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix} \tag{4.3}$$

With the coordinates of 3D points in the reference system of the left camera

$$(X_i, Y_i, Z_i) \tag{4.4}$$

we can write

$$\mathbf{A}\mathbf{x} = \mathbf{B} \tag{4.5}$$

and solve for $\mathbf{x}$ in the least-squares sense as

$$\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{B} \tag{4.6}$$

where

$$\mathbf{A} = \begin{bmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ \vdots & \vdots & \vdots \\ X_n & Y_n & Z_n \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix}$$

Often times the ground surface leads to inordinate amounts of outliers, due in part to a lack of texture from the pavement or other drivable surfaces. With the sensitivity of least-squares to outliers being known, we resort to the use of RANSAC in selecting the inliers and obtain a robust estimation of the ground plane coefficients, in the following way:

1. randomly select three points from the 3D points believed to be representative of the ground plane

2. compute the coefficients of the plane defined by the randomly selected points using (4.5)

3. count the points whose distance to the plane is less than a threshold ϵ

4. repeat these steps n times where n is sufficiently large[1]

[1] Choosing $n > 20$ does not significantly improve the number of inliers with respect to ϵ.

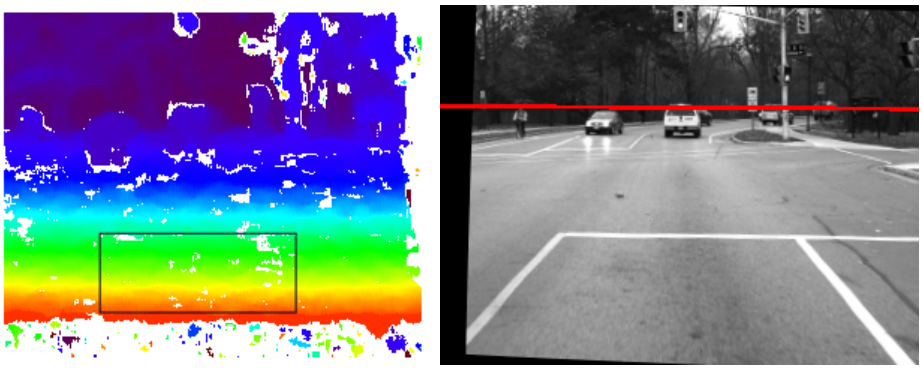

Figure 4.1: **a) (left)***: Color-coded stereo depth map and the region used for ground plane estimation* **b) (right)***: Horizon line estimated from ground plane*

5. among the n fits choose the largest inlier set which respect to ϵ and compute the coefficients of the ground plane this time using least-squares as in (4.6)

The plane parameters are averaged over a short period of time in order to stabilize them further. The coefficients of the plane are recomputed at each new stereo frame arrival. However, in cases when the number of depth values is low (poor texture, etc.) or other vision modules indicate the presence of a near obstacle, the coefficients of the ground plane are not recomputed, the previous parameters are used instead.

The horizon line is approximated by intersecting the plane parallel to that of the ground and passing through the focal point with the image plane of the sensor, and converting to image coordinates using calibration parameters. Figure 4.2 depicts the geometry involved in approximating the horizon line, while Figure 4.1 displays a sample disparity map and estimated horizon line, used in our hypothesis generation mechanism.

We generate hypotheses as rectangular shapes in which vehicles may be

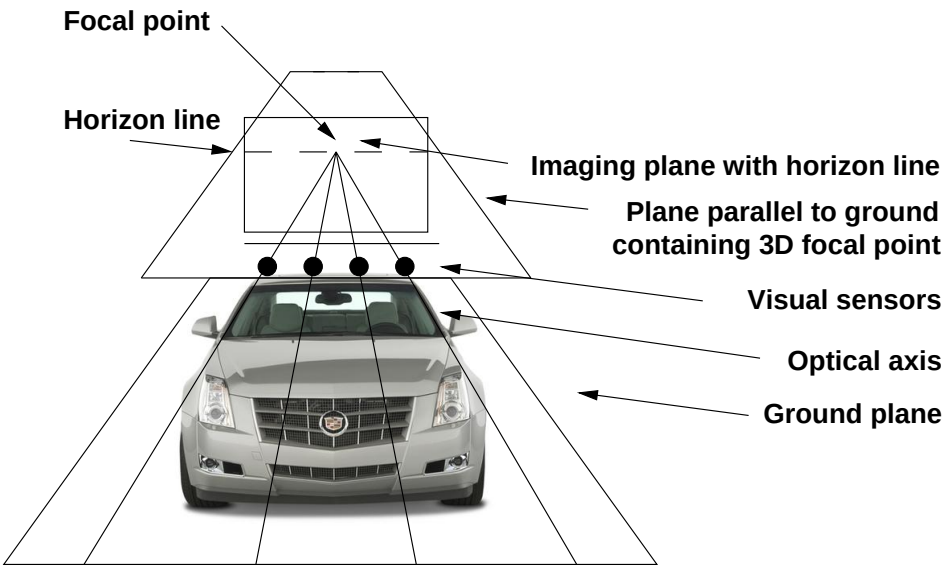

Figure 4.2: *The geometry of estimating the horizon line in camera coordinates*

found. Consider a hypothesis

$$H(x, y, w, h) \tag{4.7}$$

where x and y are the image coordinates of the top-left corner of the rectangle and w and h are the width and height of the rectangle forming the hypothesis.

The method to generate hypotheses first postulates that, since the sensors are located on the roof of the vehicle, every other vehicle with similar (or smaller) height will appear under the horizon line in the image. Consequently, the imaging of taller vehicles such as transport trucks and buses will include image sections that are above the imaged horizon line.

The fact that the horizon line is parallel to the detected ground plane does not signify that the resulting imaged horizon line is parallel to the x-axis of the image (due to the roll of the vehicle, for instance). Vehicular acceleration and deceleration generally cause tilting of the vehicle and as a consequence

the estimation of the horizon line may be somewhat imprecise at times. For this reason, we generate hypotheses for the presence of other vehicles within $\pm\delta$ vertical pixels of the imaged horizon line.

Given these observations, we generate hypotheses along a band of pixels comprised within the set of image lines $\{L\} : f(x) = mx + b \pm \delta$. For the smallest image area considered for hypothesis generation, we posit that there could be a vehicle in any of the image regions with all the possible x-axis coordinates acting as upper-left corners within the image band. For taller vehicles, we allow every generated hypothesis to grow above $l \in L$ by as much as $\frac{2}{3}$ of its own side height h. A total of four classifiers are defined and trained for the detection of

1. front views
2. back views
3. front-side and back-side views
4. side views

with some of them requiring different aspect ratios[1]. For rear and front views, the smallest hypothesis rectangle is of size 15×15 pixels, 15×36 pixels for front-side and rear-side views, and 15×45 pixels for side views. Each generated smallest-area hypothesis is then used to generate other, larger hypotheses to account for the detection of vehicles that are at closer range, and thus appear larger on the image plane. For this purpose, we scale each hypothesis by a factor of 1.2 repeatedly, until the region outgrows the image plane.

[1]The aspect ratio is 1 for the front and rear classifiers, 2.4 for vehicle front side and back side classifiers and 3.0 for the side classifier.

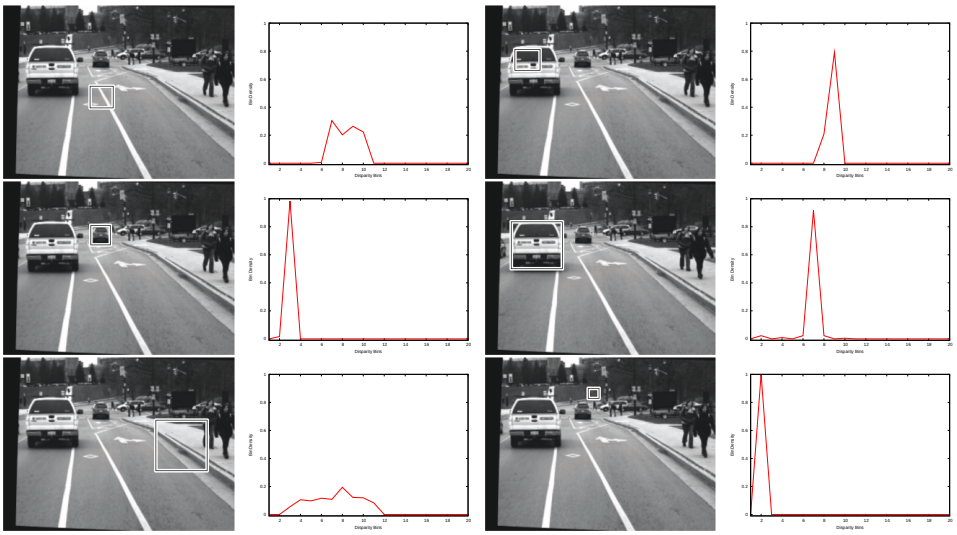

Figure 4.3: *Histogram of depth map depicted for different rectangular areas. High peaks constitute evidence of presence of an obstacle in that area. The position of the peak defines acceptable scale ranges.*

4.5 Hypothesis Verification

AdaBoost (short for Adaptive Boosting) introduced by Freund and Schapire [13] is a method for choosing and combining a set of weak classifiers to build a strong classifier. Combining the concept of the integral image as an efficient way of computing Haar-like features and cascaded AdaBoost, Viola and Jones introduced a powerful method for object recognition [38]. We adopted this approach for the hypothesis verification stage. We used four cascaded AdaBoost classifiers to discriminate positive from false-positive hypotheses. The Haar-training module from OpenCV is used to train our classifiers. We used in excess of two hundred vehicle images (positive samples) for each classifier. The images used for different vehicular views are borrowed from the dataset used by Cornells and Leibe [12, 19]. For the negative examples we used a set of more than five hundred images randomly downloaded from the Internet.

In order to increase the robustness of the hypothesis verification stage, we

considered the fact that there should be a peak in the disparity histogram where obstacles above the ground plane are imaged. Figure 4.3 shows areas with or without obstacles, and their corresponding disparity histograms. Although disparity histogram peaks have been used for obstacle detection before [16, 24], we are not aware of any work using our approach for testing several rectangular areas from the image. The main reason is probably the high computational cost of repetitive histogram generation for overlapping areas. To efficiently compute the disparity histogram, an integral disparity histogram is extracted from the disparity map. This integral disparity histogram defined by us is inspired from the integral image concept introduced by Viola and Jones [38]. To reduce the required processing time, the original disparity map is down-sampled to half of its size, and then the integral value for each bin of the histogram is computed. A total of twenty bins are used for the histogram. Computing the disparity histogram for any rectangular area amounts to performing two additions and one subtraction for each bin. The processing time required to compute the integral histogram is 5ms on conventional hardware.

Given a positive vehicle detection from our classifiers, we can further affirm that, for a given image area, if the disparity histogram peak is greater than a certain threshold, there is little probability that this hypothesis constitutes a false positive. Conversely, and again for a verified hypothesis, the absence of a clear peak is considered to signify a false-positive. Furthermore, if the peak disparity in the histogram (as a clue to obstacle depth) is inconsistent with the image area for which there is a detection, the hypothesis can be rejected outright.

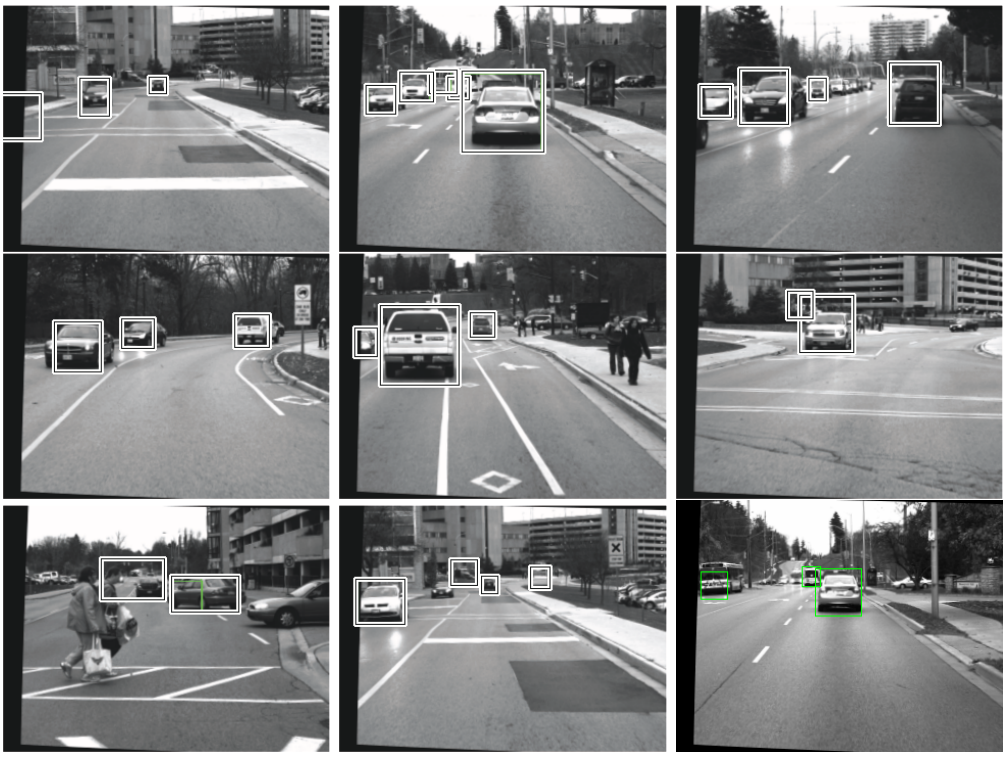

Figure 4.4: *Examples of detection results for various frames, including errors*

4.6 Hypothesis Fusing and Tracking

In every frame, image regions standing as verified hypotheses may overlap. It thus may be necessary to fuse these hypotheses into one, more consistent detection event. Hypotheses sharing more than 40% of their area are then fused into the smallest rectangular image region containing the overlapping hypotheses. This list of rectangles is used for maintaining and confirming current detections.

The representation of detected vehicles consists of a list of rectangles together with a detection counter. If this detection counter becomes higher than a certain threshold τ_2, the rectangle state changes to *active* (the probability of a vehicle in the image region is high), otherwise it remains in the list to

be either activated based on future hypothesis confirmation, or removed after sufficient evidence accumulates that no vehicle is present in it. Before new hypotheses can be added to the current list of confirmed detections, it is necessary to maintain it by tracking the detected vehicles. To this end, an optical flow field is computed for each detection along with a Kalman filter to perform the tracking.

As a new image frame becomes available, hypotheses (rectangular areas) are added to the list with their detection counter set to a value τ_1, corresponding to their maximum lifetime (measured in number of frames) before they can be removed from the list, if no confirming evidence can be found for the presence of a vehicle. We use the same technique as above to perform the merging of overlapping areas and the setting of thresholds τ_1 and τ_2. A hypothesis is removed from the list when its detection counter reaches zero. As a result, a tracking continues if a vehicle is detected every τ_1 frames on average, and a tracking is terminated if it is not detected for τ_2 frames in a row[1].

4.7 Experimental Results and Discussion

We used the RoadLAB instrumented vehicle for recording sets of sequences in the urban area of London Ontario, Canada (see Figure 4.5) [3]. The algorithm has been tested on 7,814 frames containing 13,513 vehicles in different lanes, orientations, and directions. Table 4.1 shows the accuracy of the system for vehicles closer than 50m, 100m, and 150m for *leading, oncoming, other*, and *all* vehicles, where *other* includes parked vehicles and those in other views such as at intersections. Figure 4.4 shows some correctly and incorrectly detected vehicles.

[1] Acceptable results are obtained with $\tau_1 = 3$ and $\tau_2 = 10$.

Figure 4.5: *The RoadLAB in-vehicle laboratory*

Table 4.1: Detection Rates for different distances and vehicles

Hit Rate	Leading	Oncoming	Other	All
Dis. < 50	0.9853	0.9886	0.8004	0.9858
Dis. < 100	0.8669	0.8135	0.8028	0.8548
Dis. < 150	0.8171	0.7393	0.8145	0.7990

In the sum of frames used for the experiments, the number of false positives amounts to 2,008 (about 0.26 per frame or less than 7×10^{-6} False Positive Per Window (FPPW)). Since false positives are mostly detected outside the area occupied by the roadway, better results could still be achieved by rejecting hypotheses related to detections that are not located on the roadway. The execution time of the algorithm is 25fps on generic hardware.

Our main contribution is two-fold. First, the integration of horizon-based ROI generation together with multi-view vehicle detection allows it to execute very efficiently, since many hypotheses can be rejected outright, given their relative location with respect to the horizon line. In addition, the horizon detection stage can take advantage of the tracking data and the disparity histogram to verify the absence of obstacles at close range, yielding a robust horizon line estimate. Other similar techniques often do not take advantage of detected obstacles which may lead to erroneous ground plane estimation [29, 18]. Second, the efficient integral disparity histogram-based hypothesis rejection removes a significant portion of false positives. Excellent results are obtained with AdaBoost classifiers trained on very small sets of images, resulting in a training time for each classifier under 5 minutes.

Table 4.2 shows the number of false positives and algorithm speed, using AdaBoost with and without the horizon line constraint and the depth histogram. Removing the disparity histogram constraint results in twice as many false positives, accompanied by a modest increase of the frame rate.

Figure 4.6 illustrates the difference between the detection rates for on-coming, leading, and other vehicles. Detection rates for on-coming vehicles are slightly worse than those of leading vehicles. A possible explanation may be that headlights are on in our sequences while off in the training images. The detection rates are worse for parked vehicles and those at intersections,

Table 4.2: The effect of the image horizon line and disparity histogram constraints on false positives and frame rates

	FP	FPS
Complete Technique	13%	25
Without Disparity Histogram	42%	27
Without Disparity Histogram and Horizon	58%	18

Table 4.3: Comparison on frame rates, distance (or size), hit rates, false positives, and vehicle views

Authors	Distance of Farthest Detection	FPS	HR	FP	Notes
Chang and Cho [10]	32×32	5	99%	12%	Rear detection
Southall *et al.* [30]	40m	16	99%	1.7%	Single lane rear detection
Bergmiller *et al.* [5]			83.12%	16.7%	Rear detection
Sun *et al.* [32]	32×32	10	98.5%	2%	Rear detection
Alonso *et al.* [1]			92.63%	3.63%	Rear and front detection
Cheng *et al.* [11]		20	90%	10%	Rear and front detection
Our Results	120m	25	98.6%	13%	Multi-view

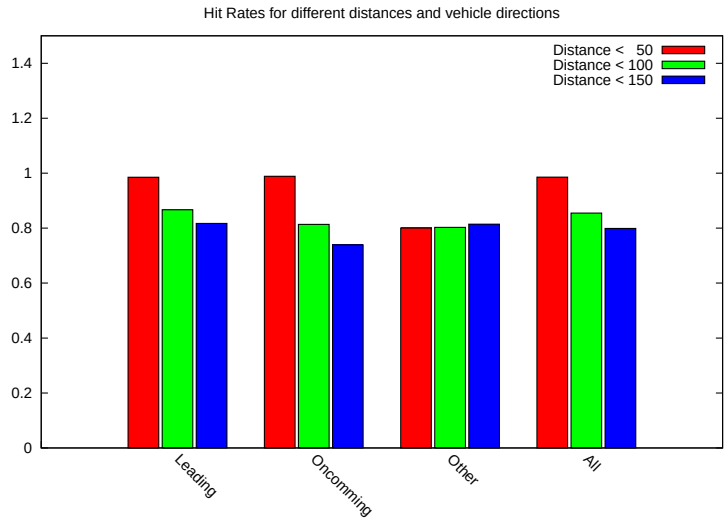

Figure 4.6: *Detection hit rates for different vehicle positions and distances*

possibly because of partial occlusion.

Figure 4.7 depicts the relation between hit-rates (provably correct detections) and vehicle distance. The performance of our technique decreases dramatically for vehicles located farther than 80m. A possible cause of this performance degradation may be that the smallest windows for which our classifiers are trained are of modest size (15×15 pixels) and provide crude resolution. We have compared the frame rates, hit rates, and the farthest detectable vehicles with other contributions cited in Table 4.3. Since all compared contributions are not applied on the same dataset, we have to trust on the fact most of the road scene and vehicles are closely comparable. Among these other techniques, ours has the best frame rate, with comparable hit rates for distances under 50m.

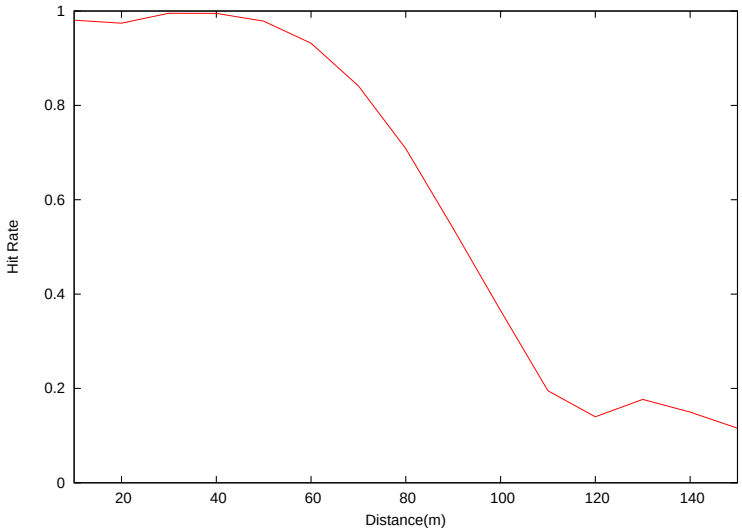

Figure 4.7: *Overall hit rate with respect to distance*

4.8 Conclusion

We have developed a real-time, multi-vehicle detection and tracking system using stereo information, optical flow, Kalman filters, and an AdaBoost classifier. The technique executes at 25fps on generic hardware and has been tested on the RoadLAB instrumented vehicle [3]. Research is currently being conducted to determine whether vehicular odometry may provide additional constraints to improve the quality of our results. We plan to use a similar framework, possibly with different constraints, to detect pedestrians in real-time.

Bibliography

[1] D. Alonso, L. Salgado, and M. Nieto. Robust vehicle detection through multidimensional classification for on board video based systems. In *Image Processing, IEEE International Conference on*, volume 4, pages 321–324, 2007.

[2] A. Amditis, K. Pagle, S. Joshi, and E. Bekiaris. Driver-vehicle-environment monitoring for on-board driver support systems: Lessons learned from design and implementation. *Applied Ergonomics*, 41:225–235, 2010.

[3] S.S. Beauchemin, M.A. Bauer, D. Laurendeau, T. Kowsari, J. Cho, M. Hunter, and O. McCarthy. RoadLab: An In-Vehicle Laboratory for Developing Cognitive Cars. In *International Conference on Computer Applications in Industry and Engineering, IEEE Conference on*, pages 7–12, 2010.

[4] A. Bensrhair, M. Bertozzi, A. Broggi, P. Miche, S. Mousset, and G. Toulminet. A cooperative approach to vision-based vehicle detection. In *Intelligent Transportation Systems*, pages 207–212, 2002.

[5] P. Bergmiller, M. Botsch, J. Speth, and U. Hofmann. Vehicle rear detection in images with Generalized Radial-Basis-Function classifiers. In *Intelligent Vehicles Symposium*, pages 226–233, 2008.

[6] M. Bertozzi and A. Broggi. GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *Image Processing, IEEE Transactions on*, 7(1):62–81, 2002.

[7] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta. Artificial vision in road vehicles. *Proceedings of the IEEE*, 90(7):1258–1271, 2002.

[8] M. Bertozzi, A. Broggi, G. Conte, and A. Fascioli. Obstacle and lane detection on the argo autonomous vehicle. In *Intelligent Transportation System, IEEE Conference on*, pages 1010–1015. IEEE, 1997.

[9] S.D. Buluswar and B.A. Draper. Color machine vision for autonomous vehicles. *Engineering Applications of Artificial Intelligence*, 11(2):245–256, 1998.

[10] W.C. Chang and C.W. Cho. Online boosting for vehicle detection. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 40(3):892–902, 2010.

[11] H. Cheng, N. Zheng, C. Sun, and H. van de Wetering. Vanishing point and gabor feature-based multi-resolution on-road vehicle detection. *ISNN 2006*, pages 46–51, 2006.

[12] N. Cornelis, B. Leibe, K. Cornelis, and L. Van Gool. 3d city modeling using cognitive loops. In *3D Data Processing, Visualization, and Transmission, 3rd International Symposium on*, pages 9–16, 2007.

[13] Y. Freund and R. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer, 1995.

[14] A. Giachetti, M. Campani, V. Torre, and C. CRS. The use of optical flow for road navigation. *IEEE transactions on robotics and automation*, 14(1):34–48, 1998.

[15] U. Handmann, T. Kalinke, C. Tzomakas, M. Werner, and W. Seelen. An image processing system for driver assistance. *Image and Vision Computing*, 18(5):367–376, 2000.

[16] H.G. Jung, Y.H. Lee, B.J. Kim, P.J. Yoon, and J.H. Kim. Stereo vision-based forward obstacle detection. *International Journal of Automotive Technology*, 8(4):493–504, 2007.

[17] T. Kalinke, C. Tzomakas, and W.V. Seelen. A texture-based object detection and an adaptive model-based classification. In *Intelligent Vehicles Symposium*, 1998.

[18] C. Keller, D. Llorca, and D. Gavrila. Dense stereo-based roi generation for pedestrian detection. *Pattern Recognition*, pages 81–90, 2009.

[19] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3d scene analysis from a moving vehicle. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, 2007.

[20] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1):259–289, 2008.

[21] C. Little. The Intelligent Vehicle Initiative: Advancing" Human-Centered" Smart Vehicles. *Public Roads*, 61(2):18–25, 1997.

[22] D.G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, page 1150. Published by the IEEE Computer Society, 1999.

[23] R. Mandelbaum, L. McDowell, L. Bogoni, B. Reich, and M. Hansen. Real-time stereo processing, obstacle detection, and terrain estimation from

vehicle-mounted stereo cameras. In *Applications of Computer Vision, 4th IEEE Workshop on*, pages 288–289, 1998.

[24] R. Mandelbaum, L. McDowell, L. Bogoni, B. Reich, and M. Hansen. Real-time stereo processing, obstacle detection, and terrain estimation from vehicle-mounted stereo cameras. In *Proceedings of the IEEE*, pages 288–289, 1998.

[25] I. Masaki. Vision-based vehicle guidance. In *Industrial Electronics, Control, Instrumentation, and Automation, Power Electronics and Motion Control, Proceedings of the International Conference on*, pages 862–867, 1998.

[26] ND Matthews, PE An, D. Charnley, and CJ Harris. Vehicle detection and recognition in greyscale imagery. *Control Engineering Practice*, 4(4):473–479, 1996.

[27] M. Maurer, R. Behringer, D. Dickmanns, T. Hildebrandt, F. Thomanek, J. Schiehlen, and E.D. Dickmanns. VaMoRs-P: An advanced platform for visual autonomous road vehicle guidance. In *Proceedings of SPIE*, volume 2352, page 239, 1995.

[28] P. Parodi and G. Piccioli. A feature-based recognition scheme for traffic scenes. In *Intelligent Vehicles Symposium, Proceedings of the*, pages 229–234, 2002.

[29] A.D. Sappa, F. Dornaika, D. Ponsa, D. Gerónimo, and A. López. An efficient approach to onboard stereo vision system pose estimation. *Intelligent Transportation Systems, IEEE Transactions on*, 9(3):476–490, 2008.

[30] B. Southall, M. Bansal, and J. Eledath. Real-time vehicle detection for highway driving. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 541–548, 2009.

[31] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection using Gabor filters and support vector machines. In *Digital Signal Processing, 14th International Conference on*, volume 2, pages 1019–1022, 2002.

[32] Z. Sun, G. Bebis, and R. Miller. Monocular precrash vehicle detection: features and classifiers. *Image Processing, IEEE Transactions on*, 15(7):2019–2034, 2006.

[33] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection: A review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, pages 694–711, 2006.

[34] Z. Sun, R. Miller, G. Bebis, and D. DiMeo. A real-time precrash vehicle detection system. In *Applications of Computer Vision, 6th IEEE Workshop on*, pages 171–176, 2003.

[35] C. Thorpe, M.H. Hebert, T. Kanade, and S.A. Shafer. Vision and navigation for the Carnegie-Mellon Navlab. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 10(3):362–373, 2002.

[36] S. Tsugawa. Vision-based vehicles in Japan: machine vision systems and driving control systems. *Industrial Electronics, IEEE Transactions on*, 41(4):398–405, 2002.

[37] B. Ulmer. Vita-an autonomous road vehicle (arv) for collision avoidance in traffic. In *Intelligent Vehicles Symposium, Proceedings of the*, pages 36–41, 1992.

[38] P. Viola and M.J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[39] J. Wu and X. Zhang. A PCA classifier and its application in vehicle detection. In *Neural Networks, International Joint Conference on*, volume 1, pages 600–604, 2002.

[40] G.W. Zhao and S. Yuta. Obstacle detection by vision system for an autonomous vehicle. In *Intelligent Vehicles Symposium*, pages 31–36, 1993.

Chapter 5

Driver Gaze Mapping

This Chapter is a reformatted version of the following article:

T. Kowsari, S.S. Beauchemin, M.A. Bauer, D. Laurendeau, and N. Teasdale, *Multi-Depth Cross-Calibration of Remote Eye Gaze Trackers and Stereoscopic Scene Systems*, accepted in IEEE Transactions on Instrumentation and Measurement, Dec. 2012.

In this Chapter we present a robust and accurate technique for the cross-calibration of 3D remote gaze trackers with stereoscopic vision systems between which no common imaging area exists. We empirically demonstrate that a multi-depth calibration approach yields remarkably superior results for obtaining 3D Point-of-Gaze (PoG) when compared with other methods using eye vergence or co-planar eye gaze calibration points.

5.1 Introduction

Remote gaze trackers have been in use for various applications together with scene cameras to determine the point of gaze (PoG) of human subjects on an imaged scene. Several types of applications benefit from the use of such sys-

tems including vehicle driver training and advanced driver assistance systems, the context in which the results herein have been obtained.

The task of projecting back the 3D gaze direction onto the imaged scene requires a cross-calibration between the remote gaze tracking device and the scene. In most if not all of commercially available systems, this type of calibration is performed by requiring that test subjects fixate specific, pre-selected image points on a planar surface placed at a known distance such as on a computer screen or, by using a scene image from a monocular camera and treating it essentially as a 2D object (co-planar fixation calibration points).

Such approaches are dependable when the subject's eye center is not highly offset from the scene camera(s). In other words, because the origin of the reference system of the scene cameras and the subject's eye center approximately coincide, the projection ray of any fixated object will also approximately lie on the line of sight regardless of the depth of the object. In such cases, the calibration process may be performed correctly. Otherwise, objects with different depths along the line of sight correspond to different image locations, and must be calibrated for as such.

Our primary goal is to determine whether driver intent and driving-related actions can be predicted from qualitative and quantitative analyses of driver behavior. Toward this end, it is necessary to establish the correspondence between cephalo-ocular behavior and visual stimuli in such a way as to identify the elements in the visual field to which driver attention turns to. This type of information in turn may facilitate the task of a driving assistance system to assess whether drivers are attending to the appropriate stimuli, given traffic context [2].

5.2 Literature Survey

Hennessey and Lawrence presented a 3D PoG method which employs eye vergence to estimate the 3D position of a fixated object [5]. In their experiments, fixated objects were contained in a 1.725 m^3 volume located in front of the subject. The reported average positional error was 3.93 cm. It constituted the first binocular system for estimating the absolute 3D coordinates of where one is looking in the 3D world.

Alternatively, Yamashiro *et al.* devised an automatic calibration to estimate the gaze of vehicle drivers by using known reference points such as the rear-view and the side mirrors of the vehicle [6]. The gaze of drivers was recorded and an Expectation-Maximization algorithm was used to cluster glances to the reference points. An automatic calibration of gaze could be achieved from the collected gaze data over time as the vehicles were driven.

In these approaches, it is assumed that the subject's eye center coincides with the origin of the reference frame of the scene cameras. When this constraint is satisfied, the depth of fixated objects does not influence the position of the gaze onto the scene images. Significant errors can be introduced otherwise, as we proceed to demonstrate in this contribution. Hence we propose a method to cross-calibrate a stereo scene camera system with a remote eye gaze tracker using variable depth calibration points and compare the resulting error with a co-planar calibration approach.

5.3 System Configuration

Our systems consist of a remote gaze tracker with two cameras pointed toward the driver's face and a stereo system oriented toward the front of an experi-

Figure 5.1: *Physical configuration* **a) (left):** *Remote eye-tracking system, and* **b) (right):** *RoadLAB stereoscopic vision system*

mental vehicle. Our aim is to determine which objects within the visual scene in front of the vehicle elicit visual responses from drivers.

Our remote gaze tracker computes several variables including gaze Euler angles, eye center location, and head position and orientation with respect to the coordinate system of the tracker, located in the middle of the stereo cameras pointed toward the driver's face. Our scene stereo system attaches to the roof of the vehicle, with its reference frame centered on the left camera. Both systems require a calibration prior to use. Figure 5.1 shows the configuration of the experimental vehicle.

5.4 Computing the Gaze Vector

The orientation of the gaze with respect to the coordinate system of the tracker is given by Euler angles describing the rotations around the X axis and the Y axis. Performing these rotations amounts to aligning the the Z axis of the tracker with the 3D direction of the gaze. The eye gaze direction defined in this way is a unit vector originating from the eye center. Figure 5.2 shows the relation between the gaze vector and a fixated point in the field of view. Given Euler angles θ_i and ϕ_i, the gaze unit vector is obtained as:

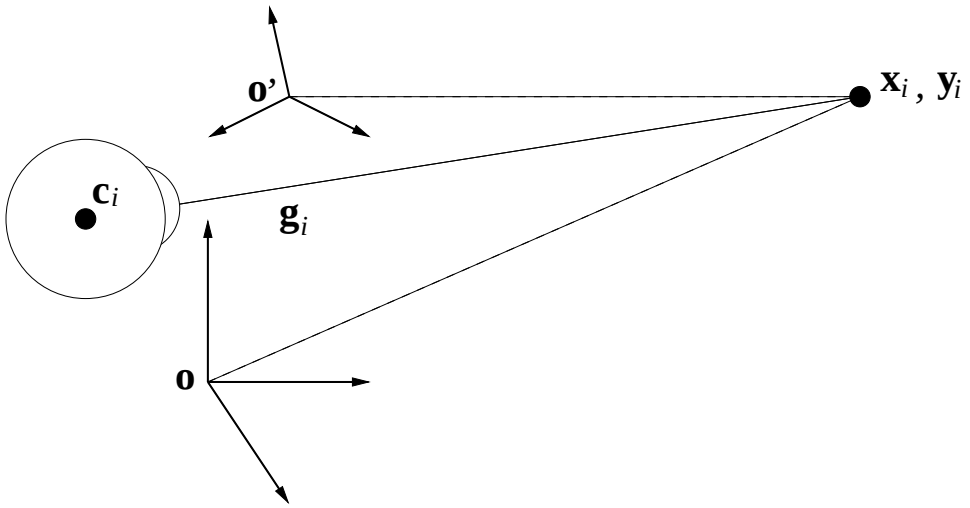

Figure 5.2: *The topology of the tracker and scene reference frames, where* $\mathbf{x}_i$ *and* $\mathbf{y}_i$ *are coordinates of the fixated point in the scene,* $\mathbf{o}$ *is the reference frame of the stereo scene system,* $\mathbf{o}'$ *that of the tracker, and* $\mathbf{c}_i$ *and* $\mathbf{g}_i$ *are the eye center position and gaze vector respectively.*

$$\mathbf{g}_i = R_y(\theta_i)R_x(\phi_i)\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \sin(\phi_i)\cos(\theta_i) \\ -\sin(\theta_i) \\ \cos(\theta_i)\cos(\phi_i) \end{bmatrix} \tag{5.1}$$

5.5 Technique

The objective consists of computing estimates of the rotation matrix and the translation vector between the reference frame of the scene stereo system and that of the remote eye tracker. The calibration process consists of asking the driver to fixate pre-selected points for which depth estimates are available and record the gaze vector and eye center location of the driver, along with the 3D position of the fixated points in the scene for a brief period (2 s) per fixated point. This data is then used to estimate the rotation matrix and the

translation vector relating the reference frames.

The eye center and gaze vector, both expressed within the reference frame of the tracker, represent a 3D line passing through the fixated point which in turn is expressed in the reference frame of the stereo scene system. Let us assume that the fixated points are known in both reference frames, and find the rigid body transformation parameters that bring the points from one reference frame to the other. The relation between the fixated points and the reference frames is given by:

$$\mathbf{y}_i = R\mathbf{x}_i + \mathbf{T} \tag{5.2}$$

where $\mathbf{x}_i$ is the position of the i^{th} fixated point measured in the scene reference frame, $\mathbf{y}_i$ is the position of $\mathbf{x}_i$ in the reference frame of the tracker, and R and $\mathbf{T}$ are the rotation matrix and translation vector between the reference frames.

We estimate the rigid transformation parameters following the approaches devised by Arun [1] and Challis [4], and use a confidence measure on the fixated points $\mathbf{x}_i$, since the accuracy of their depth measurements is inversely proportional to their distance from the stereo scene system[1].

The centers of mass of the fixated points in both reference frames are given by:

$$\bar{\mathbf{x}} = \frac{\sum_{i=1}^{n} w_i^2 \mathbf{x}_i}{\sum_{i=1}^{n} w_i^2} \quad \text{and} \quad \bar{\mathbf{y}} = \frac{\sum_{i=1}^{n} w_i^2 \mathbf{y}_i}{\sum_{i=1}^{n} w_i^2} \tag{5.3}$$

where w_i is a weight factor reflecting the reliability of the i^{th} point, and n is the number of points ($n > 2$ [4]).

[1] Consequently, we use the disparity, defined as the inverse of depth, as the confidence measure.

With the following substitution in variables:

$$\mathbf{x}'_i = \mathbf{x}_i - \bar{\mathbf{x}} \quad \text{and} \quad \mathbf{y}'_i = \mathbf{y}_i - \bar{\mathbf{y}}$$

a matrix can be formed as:

$$C = \frac{\sum_{i=1}^{n} w_i^2 \mathbf{y}'_i \mathbf{x}_i'^T}{\sum_{i=1}^{n} w_i^2} \tag{5.4}$$

and decomposed with SVD as

$$C = UDV^T \tag{5.5}$$

According to [4], setting

$$R = UV^T \tag{5.6}$$

minimizes the error in the least-squares sense. Since both the reflection and the rotation matrices minimize the least-squares error, then R is either the reflection or the rotation matrix. If R is the reflection matrix, then $\det(R) = -1$ and the rotation matrix is obtained in the following way:

$$R = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(R) \end{bmatrix} V^T \tag{5.7}$$

and the translation vector is obtained as:

$$\mathbf{T} = \bar{\mathbf{y}} - R\bar{\mathbf{x}} \tag{5.8}$$

The 3D fixated points are not directly known, as the eye center and the gaze

direction only yield a 3D line onto which a fixated point lies. Additionally, these lines are expressed within the reference frame of the eye tracker. In order to overcome this difficulty, we assume that the transformation is already known and write:

$$d_i = \|R\mathbf{x}_i + \mathbf{T}\| \tag{5.9}$$

where d_i is the distance of the i^{th} fixated point from the origin of the reference frame of the eye tracker, and $\mathbf{x}_i$ is the 3D coordinate of the point in the scene reference frame. Hence, the fixated points can be approximated in the reference frame of the tracker as:

$$\mathbf{y}_i = d_i\mathbf{g}_i + \mathbf{c}_i \tag{5.10}$$

where $\mathbf{y}_i$ is the 3D position of the fixated point in the reference frame of the tracker, with $\mathbf{g}_i$ and $\mathbf{c}_i$ defined as before. This set of fixated points allows us to approximate the rotation matrix R and the translation vector $\mathbf{T}$ iteratively. It is initially assumed that both reference frames coincide exactly, starting the iterative process with R the identity matrix and $\mathbf{T}$ a null vector. Since the reliability of the fixated points (partly) depends on their distance to the stereo system (a characterization of this error is provided in [2]), we use the stereo disparity of the fixated points, defined as the inverse of distance, to provide the weighting values w_i in (3) and (4). Algorithm 3 shows the detailed procedure for the cross-calibration.

5.6 Calibration Process

We proceed to describe the data-gathering procedure that is used with the drivers of the experimental vehicle. For each selected calibration point the

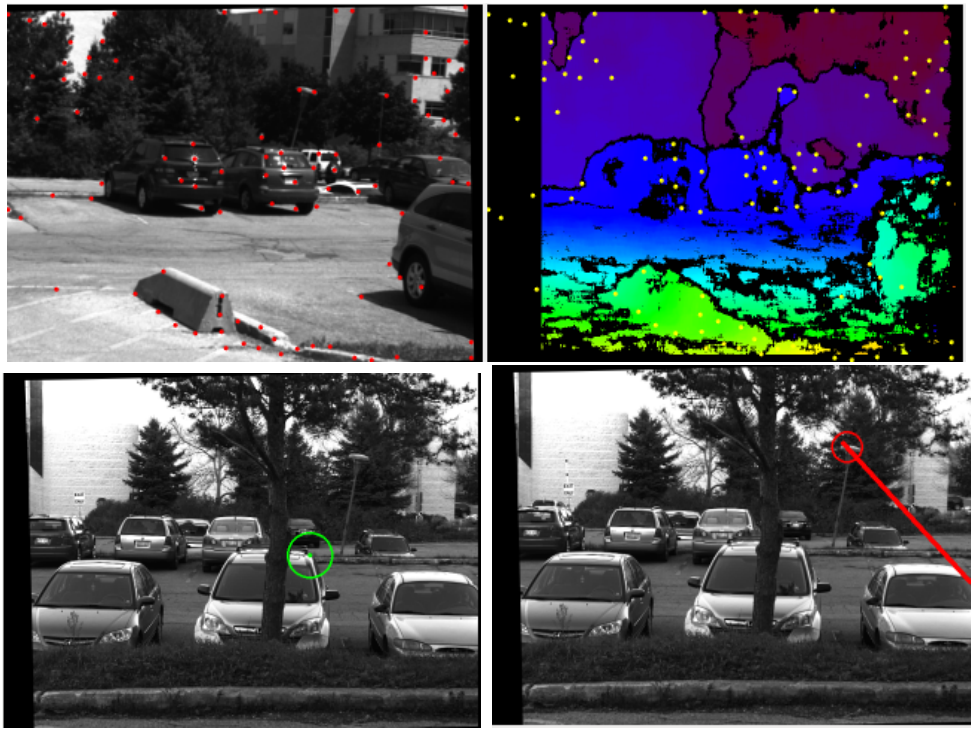

Figure 5.3: *The calibration procedure* **a) (top):** *The operator selects calibration points from a set of Hessian salient points provided by OpenCV.* **b) (bottom left):** *The driver gazes at selected points one at a time while the gaze data and depth is recorded.* **c) (bottom right):** *Driver gaze transformed into the reference frame of the stereo imaging system and intersected with the depth-map at frame rate (30Hz).*

Algorithm 3 *Cross Calibration Algorithm*

$R \leftarrow I$
$\mathbf{T} \leftarrow 0_{3\times 1}$
repeat
  for $i = 0 \rightarrow n$ **do**
    $d_i \leftarrow \|R\mathbf{x}_i + \mathbf{T}\|$
    $\mathbf{y}_i \leftarrow d_i\mathbf{g}_i + \mathbf{c}_i$
  end for
  $C \leftarrow \frac{\sum_{i=1}^{n} w_i^2 \mathbf{y}'_i \mathbf{x}_i'^T}{\sum_{i=1}^{n} w_i^2}$
  $(U, D, V^T) \leftarrow \mathrm{SVD}(C)$
  $R \leftarrow UV^T$
  $$R \leftarrow U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det(R) \end{bmatrix} V^T$$
  $\mathbf{T}' \leftarrow \mathbf{T}$
  $\mathbf{T} \leftarrow \bar{\mathbf{y}} - R\bar{\mathbf{x}}$
until $\|\mathbf{T}' - \mathbf{T}\| < \epsilon$

driver is asked to fixate, the gaze vector and the position of the eye center in the reference frame of the eye tracker are recorded, along with the 3D position of the calibration point in the reference frame of the stereo imaging system. We refer to these captured data elements as gaze data sets. While a minimum of three non co-planar calibration points are needed, we generally use 15 to 20 points to ensure sufficient precision in the computation of the calibration parameters. The calibration procedure is defined as follows:

1. Salient points provided by the stereo imaging system are detected and the calibration operator selects a suitable subset of these points (a suitable subset contains 3D points that are visible to the driver and that are found at various depths in the scene). We use the `GoodFeaturesToTrack` function from the OpenCV library to provide the initial set of salient points (Figure 5.3a).

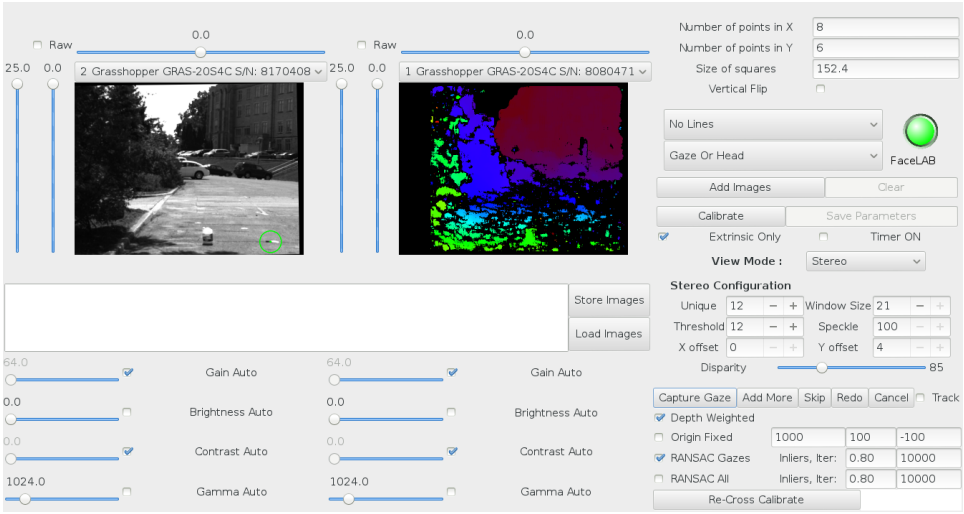

Figure 5.4: *A depiction of the integrated calibration interface. The cross-calibration area of the interface is located in the bottom right corner.*

2. The software displays the calibration point the driver is asked to fixate and records the current gaze data set for a period of 2s (Figure 5.3b). A RANSAC algorithm is used in cases when the driver experiences a saccade while requested to fixate the calibration point. This ensures the rejection of the saccade gaze data from the sample.

3. When all the points have been fixated by the driver and the gaze data recorded for each point, the operator initiates the calibration stage. Once the systems are cross-calibrated the gaze of the driver is in relation with the depth map from the stereo imaging system in real-time (see Figure 5.3a), b) and c)).

The eye tracker provides a real-time confidence measure related to the quality of the computed gaze for each eye of the driver. During the calibration process, we compute a set of cross-calibration parameters R and $\mathbf{T}$ for each eye. Once the systems are cross-calibrated and in use, we determine in real-

time which set of parameters to use based on the confidence measures provided by the eye tracker. It is possible to force the system to use a specified eye for both the calibration and the gaze projection stages in case of abnormality of one of the driver's eyes (see Figure 5.4 for a graphical representation of the calibration interface).

5.7 Projection of the Gaze on the Scene Image

Once the cross-calibration process has completed, the Line of Gaze (LoG) is projected onto the imaging plane of the stereo system and, when this line intersects with a valid depth estimate (which is most times), the PoG is then identified as the region around this intersection. To perform this projection, we first compute the 3D parameters of the LoG in the reference frame of the stereo system. The gaze vector and eye center position in the scene frame are obtained as:

$$\mathbf{g} = R^T \mathbf{g}_e \tag{5.11}$$

where $\mathbf{g}$ and $\mathbf{g}_e$ represent the gaze direction in the stereo imaging system reference system and in that of the eye tracker, respectively, and

$$\mathbf{c} = R^T(\mathbf{c}_e - \mathbf{T}) \tag{5.12}$$

where $\mathbf{c}$ and $\mathbf{c}_e$ represent the eye center position in the stereo imaging system reference system and in that of the eye tracker, respectively. Then the LoG in the scene camera coordinate system becomes

$$\frac{X - c_x}{g_x} = \frac{Y - c_y}{g_y} = \frac{Z - c_z}{g_z} \tag{5.13}$$

where $\mathbf{c} = (c_x, c_y, c_z)$ and $\mathbf{g} = (g_x, g_y, g_z)$. Using $x = \frac{X}{Z}$ and $y = \frac{Y}{Z}$ (perspective projection) yields

$$(g_x c_z - g_z c_x)y - g_x c_y = (g_y c_z - g_z c_y)x - g_y c_x \tag{5.14}$$

where x and y are 2D coordinates of the LoG stereo scene camera frame of reference. To obtain the LoG in image coordinates, the intrinsic calibration matrix of the stereo scene system is applied to the equation, resulting in

$$\begin{aligned}(g_x c_z - g_z c_x)\left(\frac{y' - s_y}{f_y}\right) - g_x c_y = \\ (g_y c_z - g_z c_y)\left(\frac{x' - s_x}{f_x}\right) - g_y c_x\end{aligned} \tag{5.15}$$

where x' and y' are image coordinates of the perspective projection of the LoG. s_x, s_y, f_x, and f_y are obtained from the intrinsic calibration matrix K of the scene stereo system:

$$K = \begin{bmatrix} f_x & 0 & s_x \\ 0 & f_y & s_y \\ 0 & 0 & 1 \end{bmatrix} \tag{5.16}$$

Then, the 2D image coordinate of the PoG is that which satisfies

$$(x'_p, y'_p) = \arg\min\left[\|Z_d - Z_l\|, \{x', y'\}\right] \tag{5.17}$$

where $\{x', y'\}$ is a pixel on the projected LoG, Z_d is its depth component, and Z_l is the corresponding depth value within the depth map. Z_d and Z_l are obtained as:

$$Z_l = \frac{c_z(g_z - g_x)}{g_z\left(\frac{x' - c_x}{f_x}\right) - g_x} \tag{5.18}$$

$$Z_d = \frac{Z}{W} \tag{5.19}$$

where Z and W originate from the re-projection in 3D of points $(x', y', d, 1)^T$:

$$\begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} = Q \begin{bmatrix} x' \\ y' \\ d \\ 1 \end{bmatrix} \tag{5.20}$$

where d is the disparity associated with (x', y') and Q is the re-projection matrix obtained with the `StereoRectify` function from OpenCv:

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -T_x^{-1} & (c_x - c'_x)T_x \end{bmatrix} \tag{5.21}$$

where (c_x, c_y) is the principal point in the left image, and c'_x the x coordinate of that of the right image [3]. Since the correct disparity d_p is immediately available once (x'_p, y'_p) is obtained with (5.17), then the 3D PoG is directly given by:

$$\mathbf{G} = (X_p, Y_p, Z_p, 1)^T = W_s^{-1}(X_s, Y_s, Z_s, W_s)^T \tag{5.22}$$

where

$$\begin{bmatrix} X_s \\ Y_s \\ Z_s \\ W_s \end{bmatrix} = Q \begin{bmatrix} x'_p \\ y'_p \\ d_p \\ 1 \end{bmatrix} \tag{5.23}$$

5.8 Experimental Protocol

Two important aspects of this technique need to be evaluated. First, an empirical convergence study must be conducted[1] and second, an error analysis performed within the conditions in which the experimental vehicle is used. We performed the convergence rate and error analysis with a group of four test drivers, composed of two males and two females, averaging 26.5 years of age. This group was composed of one Caucasian and three Middle-Eastern subjects and had no known visual problems.

5.8.1 Convergence Rate

Our study of the convergence rate begins with initializing the cross-calibration parameters R and $\mathbf{T}$. The rotation matrix is set to identity, while the translation vector is given a manually measured (an therefore approximate) vector between the centers of projection of both the eye tracker inside the vehicle and the stereo scene system on its rooftop. Figure 5.5 shows the progression of $\|\mathbf{T}' - \mathbf{T}\|$ toward 0 with respect to the number of iterations. As it is observed, a few hundred iterations (≈ 500) ensure adequate convergence for all test subjects. Since the algorithm is numerically simple, convergence is achieved within 1 s. Interestingly, convergence is particularly rapid for two of the four subjects. While only a conjecture, we believe this may be due to an unusual precision of the gaze of the test subjects when requested to fixate calibration points.

[1] Our algorithm is a straightforward extension to Arun *et al.*'s and consequently subjected to identical noiseless and noisy degenerate cases [1], justifying our decision to only study the numerical convergence rate.

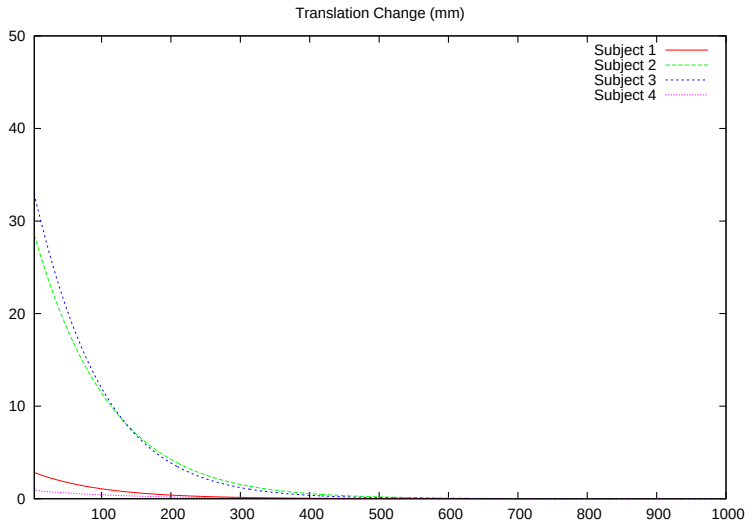

Figure 5.5: *A depiction of the convergence rate of the cross-calibration algorithm for the four test subjects.*

5.8.2 Error Analysis

The error analysis we conducted included two distinct scenarios: one for which the cross-calibration points were co-planar (CoP), and the other for which the points experienced significant non co-planarity (NcP). The aim was to compare the effects on precision when the scene camera is monocular (and hence the calibration must proceed with forcibly co-planar image points, an assumption only valid when the centers of projection of the scene camera and the eye tracker coincide) and our technique. Figure 5.6 shows a typical CoP scene along with a an NcP scene, each used for CoP and NcP calibrations, respectively. In each scenario, we measured angular error for fixated points within the scene used for calibration (which we refer to as the training scene), and then within an altogether different scene (which we refer to as the test scene), using identical cross calibration parameters for both the training and test scenes.

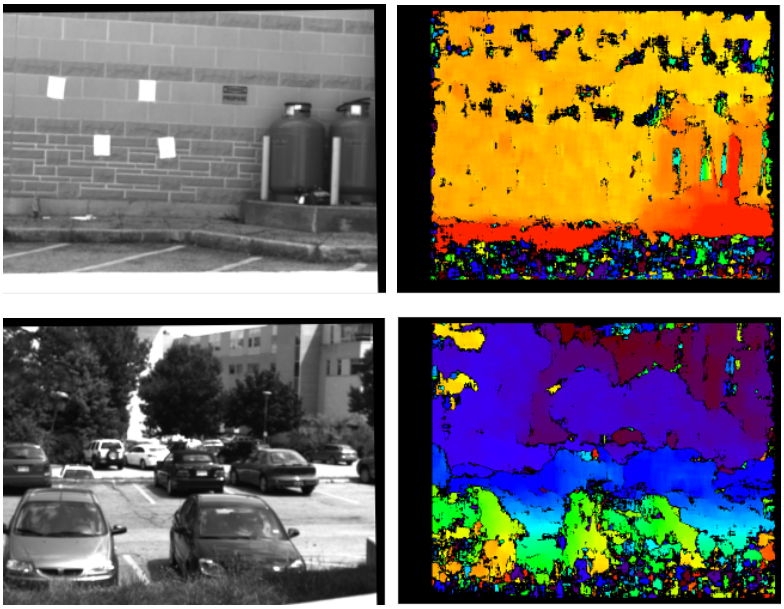

Figure 5.6: **a) (left):** *A wall and its depth map are used to perform experiments with co-planar calibration points.* **b) (right):** *A typical scene used to perform experiments with non co-planar calibration points.*

In all cases, we performed angular error analysis by requesting test subjects to fixate pre-selected points $\mathbf{p}$ in the scene for which the 3D position is known within the error margin of the scene stereo system. For each point $\mathbf{p}$, we requested the test subject to fixate it for 2 s (using the same technique as when calibrating), recorded the gaze data set, and computed its LoG in 3D, where we measured the angle between it and the LoG of $\mathbf{p}$. This method of error evaluation comprises the stereo scene system error (characterized in [2], the eye tracker error (characterized by the manufacturer of the eye tracker[1]), and whether the test subject is accurately fixating the point (difficult to quantify).

Figure 5.7a) displays the angular errors obtained on a per test subject basis. The green bars represent angular errors for the test scenes and the blue bars those from the training scenes, for both co-planar (CoP) and non co-planar

[1]FaceLAB 5, from SeeingMachines Inc.

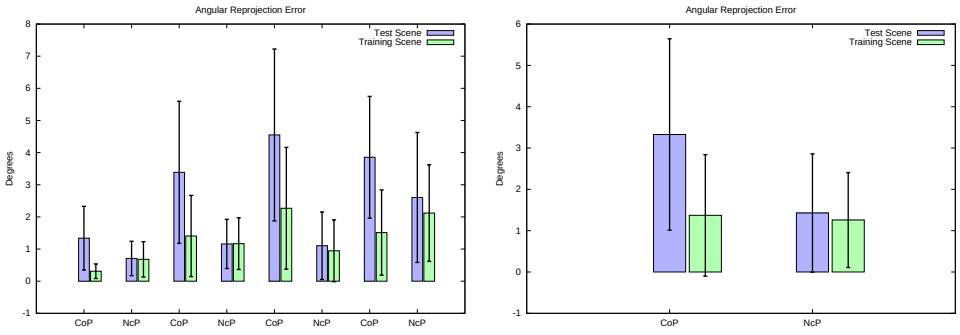

Figure 5.7: **a) (left):** *Angular errors (with standard deviation bars) obtained with the four test subjects on test and training scenes with CoP and NcP calibrations.* **b) (right):** *Angular error averages (with standard deviation bars) over the test subjects obtained on test and training scenes with CoP and NcP calibrations.*

(NcP) calibration points. As expected, for experiments conducted with CoP calibration, the errors for the test scenes (blue bars) are significantly higher than those of the training scenes (green bars). This experimental context clearly shows the inadequacy of assuming coinciding projection centers for the scene camera and the eye tracker. In the case of experiments conducted with NcP calibration, the angular error differences between the training and test scenes are significantly smaller, empirically demonstrating the superiority of our approach. This result is also clearly observed in Figure 5.7b), where errors are averaged over the test subjects. The difference in angular error between CoP and NcP calibration for the test scenes is superior to 2°, (or by a multiplicative factor just under 3).

In order to visually appreciate the error differences between CoP and NcP calibrations, we requested one of the test drivers to fixate a number of pre-selected points on a test scene with a CoP calibration (Figure 5.8a)) and then another set of pre-selected points on a test scene with an NcP calibration (Figure 5.8b)). We then projected the difference of PoGs between points requested to be fixated (displayed in green) and points actually fixated (displayed in red)

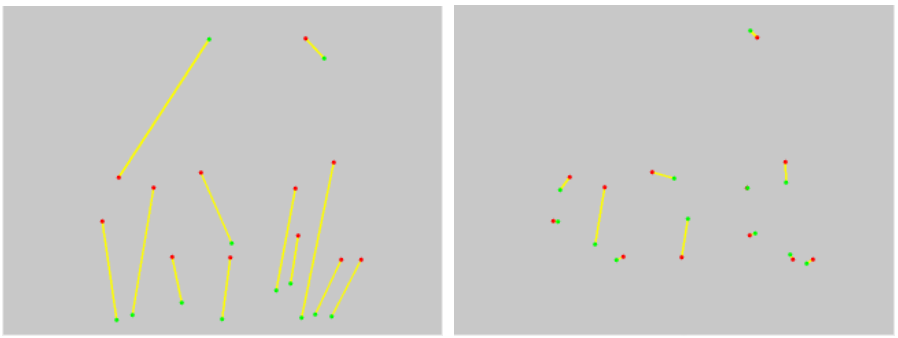

Figure 5.8: **a) (left):** *2D image errors in re-projection between points requested to be fixated and points actually fixated under a CoP calibration for a test scene* **b) (right):** *2D image errors in re-projection between points requested to be fixated and points actually fixated under an NcP calibration for a test scene*

determined by the cross calibration parameters. These results speak for themselves.

5.9 Conclusion

In 2009, Hennessey and Lawrence claimed to be first in devising a binocular system for estimating the absolute 3D coordinates of where one is looking in the 3D world, by using vergence [5]. In their experiments, fixated objects were close to test subjects and contained in a 1.725 m^3 volume. They obtained an average PoG error of 3.93 cm. We devised a novel, superior method which remains precise for much larger volumes and distances by combining a binocular eye gaze tracker with a binocular scene stereo system through an innovative cross calibration procedure. Our system operates in real time (30Hz) and is installed in an operational, experimental vehicle. To our knowledge, this experimental vehicle is the first of its kind, capable of computing the absolute 3D PoG of its driver sufficiently precisely to conduct scientific experiments addressing ocular behavior in relation to visual stimuli; an important

step toward understanding visual attention behavior and possibly predicting imminent maneuvers.

Bibliography

[1] K.S. Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, 1987.

[2] S.S Beauchemin, M.A. Bauer, T. Kowsari, and J. Cho. Portable and scalable vision-based vehicular instrumentation for the analysis of driver intentionality. *IEEE Transactions on Instrumentation and Measurement*, 61(2):391–401, 2012.

[3] G. Bradski and Kaehler A. *Learning OpenCV*. O'Reilly Media, 2008.

[4] J.H. Challis. A procedure for determining rigid body transformation parameters. *Journal of Biomechanics*, 28(6):733–737, 1995.

[5] G. Hennessey and P. Lawrence. Non-contact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions. *IEEE Transactions on Biomedical Engineering*, 8(3), 2009.

[6] K. Yamashiro, D. Deguchi, T. Takahashi, I. Ide, H. Murase, K. Higuchi, and T. Naito. Automatic calibration of an in-vehicle gaze tracking system using driver's typical gaze behavior. In *Intelligent Vehicles Symposium*, pages 998–1003, 2009.

Chapter 6

Conclusion and Future Work

Probably the most promising breakthroughs in vehicular safety will emerge from intelligent, Advanced Driving Assistance Systems (i-ADAS). Research institutions and large manufacturers work in lockstep to create advanced, on-board safety systems by means of integrating the functionality of existing systems and developing innovative sensing technologies. In this thesis, we described a portable and scalable vehicular instrumentation designed for on-road experimentation and hypothesis verification in the context of studying driver intent.

In Chapter 3, we proposed a map-based lane detection approach which robustly detects road lanes. While there exists a considerable amount of research on the subject of lane detection, we required a robust algorithm capable of using surrounding information to detect multiple lanes in various context such as rural roads, urban roads, and highways.

Subsequently, we presented an algorithm for finding the obstacle-free and safely driveable zone within lanes, based on the stereo depth map of the scene. Our experiments showed that our method robustly detects lanes and driveable

zones, even in the case of partially occluded or worn-off lane markers.

Important surrounding objects that we needed to detect and annotate in the sequences were vehicles. In Chapter 4, we proposed a multi-view, real-time vehicle detection and tracking system using stereo vision, multi-view Ada-Boost detectors, and optical flow. By adopting a ground plane estimate extracted from stereo information, we generated a sparse set of hypotheses and applied trained Ada-Boost classifiers along with disparity histogramming for Hypothesis Verification (HV) purposes. Our tracking system employed one Kalman filter per detected vehicle and motion vectors from optical flow, as a means to increase its robustness. An acceptable detection rate with few false positives was obtained at 25 Hz with generic hardware.

Lastly, in order to study drivers' visual attention with respect to the visible scene, we needed to cross-calibrate the eye-tracker with the front stereo system to obtain the precise location of the driver 3D gaze into the stereo depth map. Chapter 5 describes the algorithm and subsequent experiments that demonstrate the adequacy of the chosen approach.

The most important part of the results, other than the instrumented vehicle itself, is a set of sequences recorded from 16 subjects in real driving environments within the city of London Ontario. Each of the sequences is approximately one hour long and contains different channels of information including GPS data, vehicle odometry, driver gaze and head information, stereo depth maps, and wide angle views of the front and rear of the vehicle. After recording the sequences, an intensive effort has been made to apply the lane detection algorithm to the sequences to create a set of annotated, single channel stream of processed information.

Our contribution to this research can be summarized as follows:

1. Designing and implementing of an in-vehicle laboratory for studying driver intent

2. Presenting a novel method for multi-depth gaze cross-calibration and projection

3. Constructing a real-time vehicle detection system

4. Developing a map-based lane detection system as a novel and robust lane detection methodology.

5. Recording several sequences from drivers in natural driving environments with real vehicles on real roads, including synchronized streams of stereo images, driver gaze, GPS data, and vehicle states.

6. Annotating the sequences, and converting several heterogeneous input streams to a single stream of numeric data.

6.1 Future Work

Research on driver intent is relatively recent with the potential for significant results and applications in the near future. Here are a few possible research areas that may be undertaken immediately:

1. The main purpose of this research was to study driver intent. With the experimental data in hand and by training a predictive system, one may successfully develop a driver intent prediction system.

2. Instrumented vehicle may exchange data via V2V, and our instrumentation could be duplicated for the study of distributed ADAS.

3. While the instrumentation represents a successful proof of concept, it was noted that wider viewing angles for the stereo cameras and eye-trackers using more than two cameras (to compensate for head rotations) would allow us to track the 3D driver gaze into the surroundings in a more comprehensive manner.

4. The physical limitations of the instrumentation prevented its use at night and in adverse weather conditions. Such limitations could be removed entirely by a judicious choice of hardware, enabling the study of driver intent in diverse conditions.

Vita

NAME:	Taha Kowsari
PLACE OF BIRTH:	Eghlid Iran
YEAR OF BIRTH:	1978
POST-SECONDARY EDUCATION AND DEGREES:	University of Shiraz Shiraz, Iran 1995-1999 B.Sc. in Computer Eng.
	University of Tehran Tehran, Iran 1999-2001 M.Sc. in Computer Eng.
AWARDS:	Monash University graduate awards MFRS and MGS
	NoAE INNOVATION AWARD 2010, awarded by the NETWORK of AUTOMOTIVE EXCELLENCE (NoAE) for the RoadLAB Initiative.
	WGRS scholarship for 2008-2012, University of Western Ontario
RELATED WORK EXPERIENCE:	Software Developer Union Gas Co. 2013 - date
	Lecturer University of Western Ontario 2011-2012
	Teaching Assistant University of Western Ontario 2008 - 2011

Publications:

Kowsari T., Beauchemin S.S., Bauer M.A., "A Robust Gaze Calibration with Stereo Scene Camera ", IEEE Transactions on Instrumentation and Measurement.
Kowsari T., Beauchemin S.S., and Cho J., "Real-time Vehicle Detection and

Tracking Using Stereo Vision and Multi-View AdaBoosts", 14th International Conference on Intelligent Transportation Systems(ITSC11), Washington DC, USA, Oct. 5-7 2011

Beauchemin S.S., Bauer M.A., Kowsari T., and Cho J., "Portable and Scalable Vision-Based Vehicular Instrumentation for the Analysis of of Driver Intentionality", (in press) IEEE Transactions on Instrumentation and Measurement, Apr. 2011.

Beauchemin S.S., Bauer M.A., Laurendeau D., Kowsari T., Cho J., Hunter M., and McCarthy O., "RoadLab: An In-Vehicle Laboratory for Developing Cognitive Cars", 23rd International Conference on Computer Applications in Industry and Engineering (CAINE 10), pp. 7-12, Las Vegas, USA, Nov. 8-10 2010 (also as poster at Honda Initiation Grant Symposium 2010, Columbus. Ohio, USA, July 8 2010).

F. Noroozi, A. H. Jahangir, T. Kowsari, "Accelerate Packet Classification in Linux kernel", International Conference on Information Networking (ICOIN 2004), Pusan, Korea, Feb. 18-20, 2004

M.M. Dehghan Banadaki, M. Nili AhmadAbadi and T. Kowsari, Passive Wheel Dead-Reckoning and Gyro-Odometry Information Fusion For Mobile Robot Localization, Journal of Faculty of Engineering (special issue : Electrical and Industrial Engineering) University of Tehran,Vol. 38, No. 1, May 2004.

Taha Kowsari, Majid Nili Ahmadabadi, Hamid Soltanian Zadeh, A Real-Time Obstacle Detection Algorithm Using Optical Flow for Visual Navigation, The 11th International Conference on Advanced Robotics June 30 - July 3, 2003 University of Coimbra, Portugal.

Taha Kowsari, A Neural Network Based Edge Detection Method, proceedings of 3rd symposium in intelligent systems, 8. April 2000, University of Tehran.